CLAIMS:

1. A synchronicity independent, inter-resource and intra-processor execution-instruction delegating processor apparatus, comprising:

a processing resource;

an execution-instruction signal router, wherein the instruction signal router is disposed in communication with the processing resource;

an other processing resource, wherein the other processing resource is disposed in communication with the execution-instruction signal router;

wherein the processing resource delegates processing execution-instruction signals to an other processing resource through the execution-instruction signal router.

2. The apparatus of claim 1, wherein a processing resource is an integer processing unit.

3. The apparatus of claim 1, wherein a processing resource is a mathematical processing unit.

4. The apparatus of claim 1, wherein a processing resource is a memory management unit.

5. The apparatus of claim 1, wherein a processing resource is a vector processing unit.

6. The apparatus of claim 1, wherein a processing resource is a digital signal processing unit.

7. The apparatus of claim 1, wherein a processing resource is a graphics processing unit.

8. The apparatus of claim 1, wherein a processing resource is an input/output controller processing unit.

9. The apparatus of claim 1, wherein a processing resource is an execution-instruction processing cache.

10. The apparatus of claim 1, wherein the execution-instruction signal router is a cross-point switch.

11. The apparatus of claim 1, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

12. The apparatus of claim 1, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

13. The apparatus of claim 1, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

14. The apparatus of claim 1, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

15. The apparatus of claim 1, wherein processing resources are communicatively disposed on a same die.

16. The apparatus of claim 15, wherein the execution-instruction signal router is on the same die with processing resources.

17. A synchronicity independent, inter-resource and intra-processor execution-instruction delegating processor apparatus, comprising:

a plurality of processing resources;

a memory, wherein the memory is communicatively accessible by the processing resources, wherein the memory may be accessed simultaneously by the processing resources, wherein the memory includes an instruction unit;

wherein the plurality of processing resources and memory are on the same die.

18. The apparatus of claim 17, wherein a processing resource is an integer processing unit.

19. The apparatus of claim 17, wherein a processing resource is a mathematical processing unit.

20. The apparatus of claim 17, wherein a processing resource is a memory management unit.

21. The apparatus of claim 17, wherein a processing resource is a vector processing unit.

22. The apparatus of claim 17, wherein a processing resource is a digital signal processing unit.

23. The apparatus of claim 17, wherein a processing resource is a graphics processing unit.

24. The apparatus of claim 17, wherein a processing resource is an input/output controller processing unit.

25. The apparatus of claim 17, wherein a processing resource is an execution-instruction processing cache.

26. The apparatus of claim 17, wherein the execution-instruction signal router is a cross-point switch.

27. The apparatus of claim 17, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

28. The apparatus of claim 17, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

29. The apparatus of claim 17, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

30. The apparatus of claim 17, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

31. The apparatus of claim 17, wherein processing resources are communicatively disposed on a same die.

32. The apparatus of claim 31, wherein the execution-instruction signal router is on the same die with processing resources.

33. The apparatus of claim 17, wherein the instruction unit an instruction determination unit.

34. The apparatus of claim 17, wherein the instruction unit an instruction execution unit.

35. The apparatus of claim 17, wherein the plurality of processing resources and the memory are communicatively connected through an execution-instruction signal router.

36. A processing cache memory apparatus, comprising:

a memory;

an instruction determination unit,

wherein the memory is disposed in communication with the instruction determination unit;

wherein the instruction determination unit examines supplied signals to identify instructions and stores values into a location in the memory within a single cycle;

an instruction execution unit,

wherein the memory is disposed in communication with the instruction execution unit;

wherein the instruction execution unit executes any identified instructions and stores values into a location in the memory within a single cycle.

37. The apparatus of claim 36, wherein the instruction determination unit and the instruction execution unit are a singular unit.

38. The apparatus of claim 36, wherein the memory, instruction determination unit, instruction execution unit are on the same die.

39. The apparatus of claim 36, wherein the instruction determination unit and the instruction execution units are capable of determining and executing an event-sleep command.

40. The apparatus of claim 36, wherein the instruction determination unit and the instruction execution units are capable of determining and executing an event-set command.

41. The apparatus of claim 36, wherein the instruction determination unit and the instruction execution units are capable of determining and executing a wait-on-semaphore command.

42. The apparatus of claim 36, wherein the instruction determination unit and the instruction execution units are capable of determining and executing a signal-on-semaphore command.

43. The apparatus of claim 36, wherein the instruction determination unit and the instruction execution units are capable of determining and executing a sleep-lock command.

44. The apparatus of claim 36, wherein the instruction determination unit and the instruction execution units are capable of determining and executing a sleep-lock-destruction command.

45. The apparatus of claim 36, wherein the instruction determination unit and the instruction execution units are capable of determining and executing a add command.

46. The apparatus of claim 36, wherein the instruction determination unit and the instruction execution units are capable of determining and executing a logical-OR command.

47. The apparatus of claim 36, wherein the instruction determination unit and the instruction execution units are capable of determining and executing a logical-AND command.

48. The apparatus of claim 36, wherein the instruction determination unit and the instruction execution units are capable of determining and executing a logical-NOT command.

49. The apparatus of claim 36, wherein a processing resource may sleep while an other processing resource executes execution-instructions.

50. The method of claim 49, wherein a sleeping processing resource may be woken by a response to a request.

51. The method of claim 50, wherein the response is directed to a specific processing resource.

52. The method of claim 50, wherein the response indicates a once locked resource is unlocked.

53. The apparatus of claim 49, wherein the other processing resource is the memory.

54. The apparatus of claim 36, wherein a plurality of processing resources and the memory are communicatively connected through an execution-instruction signal router.

55. An inter-resource, intraprocessor execution-instruction delegation apparatus, comprising:

a resource arbitration unit,

wherein the resource arbitration unit selects a source processing resource from which to obtain an execution-instruction signal,

wherein the resource arbitration unit includes a priority bit comparator,

wherein the priority bit comparator is communicatively disposed with the resource arbitration unit

wherein the resource arbitration unit is communicatively disposed with processing resources.

56. The apparatus of claim 55, wherein the arbitration unit is a counter employing $\log_2$ bits a number of total processing resources to address processing resources.

57. The apparatus of claim 55, wherein the number of total processing resources is a number of processing resources interconnected through the resource arbitration unit within a single die.

58. The apparatus of claim 55, wherein the number of total processing resources is a number of processing resources within a single processor.

59. The apparatus of claim 55, wherein the resource providing an execution-instruction signal with the highest priority is selected by the resource arbitration unit.

60. The apparatus of claim 55, wherein the resource arbitration unit and the priority bit comparator obtain instructions and are communicatively disposed.

61. The apparatus of claim 55, further comprising,

an iterating dead-lock bit counter.

62. The apparatus of claim 61, wherein the resource arbitration unit selects a source processing resource's execution-instruction signal by employing the dead-lock bit counter.

63. The apparatus of claim 55, further comprising,

a resource delegation unit.

64. The apparatus of claim 63, wherein the resource delegation unit is communinicatively disposed with the resource arbitration unit.

65. The apparatus of claim 64, wherein the resource delegation unit directs a selected execution-instruction signal to a processing resource.

66. The apparatus of claim 64, wherein the resource delegation unit employs the selected execution-instruction signal's operation-code to identify a target processing resource.

67. The apparatus of claim 55, wherein the execution-instruction signal was selected by the resource arbitration unit.

68. The apparatus of claim 55, wherein the resource arbitration unit is a cross-point switch; wherein the resource arbitration unit has less than a 100 picosecond delay.

69. The apparatus of claim 55, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions through the resource arbitration unit.

70. The apparatus of claim 55, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

71. The apparatus of claim 55, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

72. The apparatus of claim 55, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

73. The apparatus of claim 55, wherein the resource arbitration unit is on the same die with processing resources.

74. The apparatus of claim 55, wherein a plurality of processing resources and a memory are communicatively connected through a resource arbitration unit.

75. A medium readable by a processor, comprising:

execution-instruction signals in the processor readable medium, wherein the execution-instruction signals are issuable by the processor and include:

a processing resource identifier, wherein the processing resource identifier identifies the origin of an execution-instruction signal;

an operation-code, wherein the operation code identifies a target processing resource;

a data field, wherein the data field may be used by a processing resource as an operand;

a priority field, wherein the priority field may be used to prioritize scheduling of an execution-instruction signal.

76. The medium of claim 75, wherein the processing resource identifier is log2 bits the number of available processing resources.

77. The medium of claim 75, wherein the operation-code is 10 bits.

78. The medium of claim 75, wherein the data field may contain an address.

79. The medium of claim 75, wherein the data field may contain information to be processed.

80. The medium of claim 75, wherein the data field is 64 bits.

81. The medium of claim 75, wherein the priority field is 2 bits.

82. The medium of claim 75, further comprising,
an address field.

83. The medium of claim 82, wherein the address field is 32 bits.

84. The medium of claim 82, wherein the address field employs literal prefixes.

85. A processor comprising:
an integer processing unit adapted to execute instructions of a program;
a math processing unit adapted to receive a request from the integer processing unit; and
a router that interfaces between the integer processing unit and the math processing unit, wherein the router is adapted to route the request from the integer processing unit to the math processing unit.

86. The processor according to claim 85, wherein the integer processing unit is adapted to execute instructions only with either an add calculation or a subtract calculation of integer numbers.

87. The processor according to claim 85, wherein the integer processing unit is adapted to detect an instruction with either a floating-point calculation, a multiple calculation or a divide calculation.

88. The processor according to claim 87, wherein the integer processing unit is adapted to extract an opcode from the instruction upon detection and forward the opcode to the math processing unit with a calculation request through the router.

89. The processor according to claim 87, wherein the integer processing unit is adapted to forward the instruction to the math processing unit with a calculation request through the router.

90. The processor according to claim 85, wherein the integer processing unit includes an internal cache and the internal cache is divided into a local and global areas thereby increasing a cache hit rate.

91. The processor according to claim 85, wherein the math processing unit is adapted to receive the request using a pipeline structure.

92. The processor according to claim 85, wherein the math processing unit is adapted to return a result in response to the request from the integer unit.

93. The processor according to claim 85, further comprising a second integer processing unit adapted to execute instructions either from one program or from another program.

94. The processor according to claim 85, further comprising a second math processing unit adapted to receive a request from the integer processing unit.

95. The processor according to claim 85, further comprising a cache unit which interfaces with the integer processing unit through the router, wherein the cache unit is adapted to receive a request from the integer processing unit.

96. The processor according to claim 95, wherein the cache unit is adapted to receive the request using a pipeline structure.

97. The processor according to claim 95, wherein the cache unit is divided into a local and global areas thereby increasing a cache hit rate.

98. The processor according to claim 95, wherein the cache unit is adapted to perform small set of atomic functions including a bus-lock control and a read-modify-write control.

99. The processor according to claim 95, wherein the cache unit is adapted to be connected directly to any of a memory, a test access port and a time divisional multiplexing data port.

100. The processor according to claim 99, wherein the test access port is a port for the Joint Test Action Group (JTAG) with a standard IEEE 1149.

101. A processor comprising:

a plurality of processing units each to execute instructions of a program independently; and

a cache unit configured to receive access requests from the plurality of processing units and returns results to the plurality of processing units in response to the access requests,

wherein each of the plurality of processing units is configured to sleep after sending an access request to the cache unit.

102. The processor of claim 101, wherein the cache unit is further configured to broadcast a signal to the plurality of processing units so that the plurality of processing units may receive the signal simultaneously.

103. The processor of claim 102, wherein the signal includes a requested result from one of the plurality of processing units combined with an identification information of the one of the plurality of processing units.

104. The processor of claim 103, wherein each of the plurality of the processing units is configured to detect the identification information thereby determining whether the identification information matches with its own identification information.

105. The processor of claim 104, wherein each of the plurality of the processing units is configured to wake up and continues operation with the requested result upon determining that the identification information matches with its own identification information.

106. The processor of claim 102, wherein the signal includes an availability information of the cache unit.

107. The processor of claim 106, wherein each of the plurality of the processing units is configured to detect the availability information thereby determining whether the cache unit is available.

108. The processor of claim 107, wherein each of the plurality of processing units is configured to wake up upon determining that the cache unit is available.

109. The processor of claim 108, wherein each of the plurality of processing units is configured to send another access request to the cache unit upon determining that the cache unit is available.

110. The processor of claim 101, wherein each of the plurality of processing units is assigned with a priority order over other processing units thereby accessing the cache unit according to the priority order.

111. A processor comprising:

a processing unit with an internal cache unit, residing internal to the processing unit, configured to execute instructions of a program; and

an external cache unit, residing external to the processing unit, configured to cooperate with the internal cache unit of the processing unit,

wherein each of the internal and external cache units is divided into an instruction area and a data area, and the data area is further subdivided into a local area and a global area.

112. The processor of claim 111, wherein the processing unit is configured to copy a portion of the global area of the internal cache into the global area of the external cache after executing a portion of the instructions.

113. The processor of claim 112, wherein the portion of the instructions is either a semaphore instructions or an unlock instructions of a program.

114. The processor of claim 111, wherein the processing unit is configured to discard a portion of the global area of the internal cache.

115. The processor of claim 111, wherein the processing unit is configured to determine whether a portion of the global area of the internal cache is accessed after executing a portion of the instructions.

116. The processor of claim 115, wherein the processing unit is configured to copy the portion of the global area of the internal cache into the global area of the external cache

after executing the portion of the instructions upon determining that the portion of the global area is accessed by the processing unit.

117. The processor of claim 115, wherein the processing unit is configured to discard the portion of the global area of the internal cache after executing the portion of the instructions upon determining that the portion of the global area is not accessed by the processing unit.

118. A method of register addressing, comprising:

obtaining a register bank address;

decoding an operation-code;

setting a cycle flag in a status register based on the decoding of the operation-code;.

accessing a register addressed by the cycle flag within the addressed register bank.

119. The method of claim 118, further comprising,

employing a set number of bits in an execution-instruction to establish the number of registers within a register bank.

120. The method of claim 118, further comprising,

employing a set number of bits in an execution-instruction to establish the number of register banks.

121. A method of setting intra-processor processing resources to sleep, comprising:

setting processing resources that issue requests to delegating processing resources to sleep until a response is provided;

setting processing resources waiting on shared and locked memory being accessed by other processing resources to sleep until the memory is unlocked.

122. A method of execution-instruction delegation between processing resources, comprising:

sharing memory between processing resources

wherein the memory itself includes instruction execution logic;

wherein the processing resources are communicatively accessible through an instruction execution router;

wherein the processing resources are on the same die;

delegating execution-instructions from originating processing resources to other processing resources.

123. The method of claim 122, wherein the method is completed within a single processing cycle.

124. The method of claim 122, wherein the other processing resource may be the originating processing resource.

125. The method of claim 122, wherein a processing resource is an integer processing unit.

126. The method of claim 122, wherein a processing resource is a mathematical processing unit.

127. The method of claim 122, wherein a processing resource is a memory management unit.

128. The method of claim 122, wherein a processing resource is a vector processing unit.

129. The method of claim 122, wherein a processing resource is a digital signal processing unit.

130. The method of claim 122, wherein a processing resource is a graphics processing unit.

131. The method of claim 122, wherein a processing resource is an input/output controller processing unit.

132. The method of claim 122, wherein a processing resource is an execution-instruction processing cache.

133. The method of claim 122, wherein delegation occurs through an execution-instruction signal router.

134. The method of claim 133, wherein the execution-instruction signal router is a cross-point switch.

135. The method of claim 122, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

136. The method of claim 122, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

137. The method of claim 122, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

138. The method of claim 122, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

139. The method of claim 135, wherein a sleeping processing resource may be woken by a response to a request.

140. The method of claim 139, wherein the response is directed to a specific processing resource.

141. The method of claim 139, wherein the response indicates a once locked resource is unlocked.

142. The method of claim 122, wherein processing resources are communicatively disposed on a same die.

143. The method of claim 142, wherein an execution-instruction signal router is on the same die with processing resources.

144. A method of execution-instruction delegation between processing resources, comprising:

obtaining an execution instruction, wherein the execution instruction is obtained at a processing resource;

determining whether an operation-code within the execution instruction should be delegated to an other processing resource;

executing the execution instruction, if the operation-code within the execution instruction should not be delegated to an other processing resource;

routing the execution instruction to an other processing resource, if the operation-code within the execution instruction is for an other processing resource.

145. The method of claim 144, wherein the method is completed within a single processing cycle.

146. The method of claim 144, wherein the routed execution instruction is a request for execution by a type of processing resource.

147. The method of claim 146, wherein the request includes an operation-code.

148. The method of claim 147, wherein the operation-code indicates a type of resource on which to execute.

149. The method of claim 146, wherein the request includes values from a status register.

150. The method of claim 146, wherein the request includes values from a priority bit status register.

151. The method of claim 146, wherein the request includes a processing resource identifier.

152. The method of claim 151, wherein the processing resource identifier is an integer processing unit number.

153. The method of claim 146, wherein the request is obtained by an execution-instruction signal router.

154. The method of claim 146, wherein the request includes an address.

155. The method of claim 144, wherein the operation-code indicates a type of resource on which to execute.

156. The method of claim 144, wherein the other processing resource may be the originating processing resource.

157. The method of claim 144, wherein a processing resource is an integer processing unit.

158. The method of claim 144, wherein a processing resource is a mathematical processing unit.

159. The method of claim 144, wherein a processing resource is a memory management unit.

160. The method of claim 144, wherein a processing resource is a vector processing unit.

161. The method of claim 144, wherein a processing resource is a digital signal processing unit.

162. The method of claim 144, wherein a processing resource is a graphics processing unit.

163. The method of claim 144, wherein a processing resource is an input/output controller processing unit.

164. The method of claim 144, wherein a processing resource is an execution-instruction processing cache.

165. The method of claim 144, wherein routing occurs through an execution-instruction signal router.

166. The method of claim 165, wherein the execution-instruction signal router is a cross-point switch.

167. The method of claim 144, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

168. The method of claim 144, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

169. The method of claim 144, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

170. The method of claim 144, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

171. The method of claim 144, wherein processing resources are communicatively disposed on a same die.

172. The method of claim 171, wherein an execution-instruction signal router is on the same die with processing resources.

173. A method of execution-instruction delegation between processing resources, comprising:

obtaining an request signal, wherein the request signal is from a requesting processing resource;

determining a priority dead-lock-avoidance value, wherein the priority dead-lock value is used to select among multiple requests with equal priority;

examining a request priority value for each submitted request;

selecting a request with a highest priority value, if more than one processing resource requests a same target resource;

providing a selected request to a target processing resource.

174. The method of claim 173, wherein the method is completed within a single processing cycle.

175. The method of claim 173, wherein the request signal is a request for execution by a type of processing resource.

176. The method of claim 175, further comprising,

routing the request to an other processing resource, if an operation-code within the request is for an other processing resource.

177. The method of claim 175, wherein the request is obtained at an execution-instruction signal router.

178. The method of claim 175, wherein the execution-instruction signal is provided by a delegating processing resource via an execution-instruction signal router.

179. The method of claim 175, wherein the request includes an operation-code.

180. The method of claim 179, wherein the operation-code indicates a type of resource on which to execute.

181. The method of claim 175, wherein the request includes values from a status register.

182. The method of claim 175, wherein the request includes values from a priority bit status register.

183. The method of claim 175, wherein the request includes a processing resource identifier.

184. The method of claim 183, wherein the processing resource identifier is an integer processing unit number.

185. The method of claim 175, wherein the request includes an address.

186. The method of claim 173, wherein the other processing resource may be the originating processing resource.

187. The method of claim 173, wherein a processing resource is an integer processing unit.

188. The method of claim 173, wherein a processing resource is a mathematical processing unit.

189. The method of claim 173, wherein a processing resource is a memory management unit.

190. The method of claim 173, wherein a processing resource is a vector processing unit.

191. The method of claim 173, wherein a processing resource is a digital signal processing unit.

192. The method of claim 173, wherein a processing resource is a graphics processing unit.

193. The method of claim 173, wherein a processing resource is an input/output controller processing unit.

194. The method of claim 173, wherein a processing resource is an execution-instruction processing cache.

195. The method of claim 173, wherein delegation occurs through an execution-instruction signal router.

196. The method of claim 195, wherein the execution-instruction signal router is a cross-point switch.

197. The method of claim 173, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

198. The method of claim 173, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

199. The method of claim 173, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

200. The method of claim 173, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

201. The method of claim 197, wherein a sleeping processing resource may be woken by a response to a request.

202. The method of claim 201, wherein the response is directed to a specific processing resource.

203. The method of claim 201, wherein the response indicates a once locked resource is unlocked.

204. The method of claim 173, wherein processing resources are communicatively disposed on a same die.

205. The method of claim 204, wherein an execution-instruction signal router is on the same die with processing resources.

206. The method of claim 173, wherein the priority dead-lock-avoidance value is iterated.

207. The method of claim 173, further comprising,

selecting a request, if not more than one processing resource requests a same target resource.

208. The method of claim 173, further comprising,

selecting the request with a highest priority value, if not more than one request has a same highest priority value;

selecting a request based on the priority dead-lock-avoidance value, if more than one request has a same highest priority value.

209. The method of claim 173, further comprising,

providing a request granted acknowledgement to the requesting processing resource whose request was selected.

210. The method of claim 173, further comprising,

clearing the operation-code in the requesting processing resource whose request was selected.

211. A method of execution-instruction delegation between processing resources, comprising:

processing a request execution-instruction, wherein the request execution-instruction includes a requesting processing resource identifier;

preparing a response into a result register, wherein the response includes the requesting processing resource identifier;

presenting the response to all processing resources.

212. The method of claim 211, wherein the method is completed within a single processing cycle.

213. The method of claim 211, wherein the request execution-instruction is a request for execution by a type of processing resource.

214. The method of claim 213, further comprising,

routing the request to an other processing resource, if an operation-code within the request is for an other processing resource.

215. The method of claim 213, wherein the request is obtained at an execution-instruction signal router.

216. The method of claim 213, wherein the execution-instruction signal is provided by a delegating processing resource via an execution-instruction signal router.

217. The method of claim 213, wherein the request includes an operation-code.

218. The method of claim 217, wherein the operation-code indicates a type of resource on which to execute.

219. The method of claim 213, wherein the request includes values from a status register.

220. The method of claim 213, wherein the request includes values from a priority bit status register.

221. The method of claim 213, wherein the request includes a processing resource identifier.

222. The method of claim 221, wherein the processing resource identifier is an integer processing unit number.

223. The method of claim 213, wherein the request includes an address.

224. The method of claim 211, wherein the other processing resource may be the originating processing resource.

225. The method of claim 211, wherein a processing resource is an integer processing unit.

226. The method of claim 211, wherein a processing resource is a mathematical processing unit.

227. The method of claim 211, wherein a processing resource is a memory management unit.

228. The method of claim 211, wherein a processing resource is a vector processing unit.

229. The method of claim 211, wherein a processing resource is a digital signal processing unit.

230. The method of claim 211, wherein a processing resource is a graphics processing unit.

231. The method of claim 211, wherein a processing resource is an input/output controller processing unit.

232. The method of claim 211, wherein a processing resource is an execution-instruction processing cache.

233. The method of claim 211, wherein delegation occurs through an execution-instruction signal router.

234. The method of claim 233, wherein the execution-instruction signal router is a cross-point switch.

235. The method of claim 211, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

236. The method of claim 211, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

237. The method of claim 211, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

238. The method of claim 211, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

239. The method of claim 235, wherein a sleeping processing resource may be woken by a response to a request.

240. The method of claim 239, wherein the response is directed to a specific processing resource.

241. The method of claim 239, wherein the response indicates a once locked resource is unlocked.

242. The method of claim 211, wherein processing resources are communicatively disposed on a same die.

243. The method of claim 242, wherein an execution-instruction signal router is on the same die with processing resources.

244. A method of execution-instruction delegation between processing resources, comprising:

obtaining a result execution-instruction from a delegate processing resource, wherein the result includes a requesting processing resource identifier;

waking the requesting processing resource, if the requesting processing resource identifier identifies the instant processing resource;

waking a processing resource, if a processing resource is waiting to be unlocked and if the obtained result includes an unlock flag and if an address in the result matches a locked address of an instant processing resource.

245. The method of claim 244, wherein the method is completed within a single processing cycle.

246. The method of claim 244, wherein the result execution-instruction is an execution-instruction signal for execution by a type of processing resource.

247. The method of claim 246, further comprising,

routing the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

248. The method of claim 246, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

249. The method of claim 246, wherein the execution-instruction signal is provided by a delegating processing resource via an execution-instruction signal router.

250. The method of claim 246, wherein the execution-instruction signal includes an operation-code.

251. The method of claim 250, wherein the operation-code indicates a type of resource on which to execute.

252. The method of claim 246, wherein the execution-instruction signal includes values from a status register.

253. The method of claim 246, wherein the execution-instruction signal includes values from a priority bit status register.

254. The method of claim 246, wherein the execution-instruction signal includes a processing resource identifier.

255. The method of claim 254, wherein the processing resource identifier is an integer processing unit number.

256. The method of claim 246, wherein the execution-instruction signal includes an address.

257. The method of claim 244, wherein an other processing resource may be an delegating processing resource.

258. The method of claim 244, wherein a processing resource is an integer processing unit.

259. The method of claim 244, wherein a processing resource is a mathematical processing unit.

260. The method of claim 244, wherein a processing resource is a memory management unit.

261. The method of claim 244, wherein a processing resource is a vector processing unit.

262. The method of claim 244, wherein a processing resource is a digital signal processing unit.

263. The method of claim 244, wherein a processing resource is a graphics processing unit.

264. The method of claim 244, wherein a processing resource is an input/output controller processing unit.

265. The method of claim 244, wherein a processing resource is an execution-instruction processing cache.

266. The method of claim 244, wherein delegation occurs through an execution-instruction signal router.

267. The method of claim 266, wherein the execution-instruction signal router is a cross-point switch.

268. The method of claim 244, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

269. The method of claim 244, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

270. The method of claim 244, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

271. The method of claim 244, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

272. The method of claim 268, wherein a sleeping processing resource may be woken by a response to a request.

273. The method of claim 272, wherein the response is directed to a specific processing resource.

274. The method of claim 272, wherein the response indicates a once locked resource is unlocked.

275. The method of claim 244, wherein processing resources are communicatively disposed on a same die.

276. The method of claim 275, wherein an execution-instruction signal router is on the same die with processing resources.

277. The method of claim 244, further comprising,

executing a next execution-instruction, if the requesting processing resource identifier identifies the instant processing resource;

executing an execution-instruction that failed to execute, if a processing resource is waiting to be unlocked and if the obtained result includes an unlock flag and if an address in the result matches a locked address of an instant processing resource.

278. A method of memory access optimization, comprising:

obtaining a storage-access request from a delegated processing resource, wherein the request includes a target memory address;

determining the target memory address from the request;

comparing the target memory address with register values, wherein register values are used to establish a data type of the target memory address for subsequent storage in an apportioned region of cache memory;

obtaining information from the target memory address;

storing the information in an apportioned region of cache memory.

279. The method of claim 278, wherein the method is completed within a single processing cycle.

280. The method of claim 278, wherein the storage-access request is an execution-instruction signal for execution by a type of processing resource.

281. The method of claim 280, further comprising,

routing the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

282. The method of claim 280, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

283. The method of claim 280, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

284. The method of claim 280, wherein the execution-instruction signal includes an operation-code.

285. The method of claim 284, wherein the operation-code indicates a type of resource on which to execute.

286. The method of claim 280, wherein the execution-instruction signal includes values from a status register.

287. The method of claim 280, wherein the execution-instruction signal includes values from a priority bit status register.

288. The method of claim 280, wherein the execution-instruction signal includes a processing resource identifier.

289. The method of claim 288, wherein the processing resource identifier is an integer processing unit number.

290. The method of claim 280, wherein the execution-instruction signal includes an address.

291. The method of claim 278, wherein an other processing resource may be an delegating processing resource.

292. The method of claim 278, wherein a processing resource is an integer processing unit.

293. The method of claim 278, wherein a processing resource is a mathematical processing unit.

294. The method of claim 278, wherein a processing resource is a memory management unit.

295. The method of claim 278, wherein a processing resource is a vector processing unit.

296. The method of claim 278, wherein a processing resource is a digital signal processing unit.

297. The method of claim 278, wherein a processing resource is a graphics processing unit.

298. The method of claim 278, wherein a processing resource is an input/output controller processing unit.

299. The method of claim 278, wherein a processing resource is an execution-instruction processing cache.

300. The method of claim 278, wherein delegation occurs through an execution-instruction signal router.

301. The method of claim 300, wherein the execution-instruction signal router is a cross-point switch.

302. The method of claim 278, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

303. The method of claim 278, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

304. The method of claim 278, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

305. The method of claim 278, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

306. The method of claim 302, wherein a sleeping processing resource may be woken by a response to a request.

307. The method of claim 306, wherein the response is directed to a specific processing resource.

308. The method of claim 306, wherein the response indicates a once locked resource is unlocked.

309. The method of claim 278, wherein processing resources are communicatively disposed on a same die.

310. The method of claim 309, wherein an execution-instruction signal router is on the same die with processing resources.

311. The method of claim 278, wherein the storage-access request is obtained at a processing resource.

312. The method of claim 278, wherein information at the target memory address is designated to be local data if the target memory address is within an address range established by stack base and end register values.

313. The method of claim 278, wherein information at the target memory address is designated to be global data if the target memory address is outside an address range established by stack base and end register values.

314. The method of claim 278, wherein an apportionment is designated for each type of data for optimized access.

315. The method of claim 278, wherein the apportioned region is a hash region.

316. The method of claim 315, wherein a hash region is designated for local data.

317. The method of claim 315, wherein a hash region is designated for global data.

318. The method of claim 315, wherein the apportionment for a local data region is greater than a region for global data.

319. A method of reduced size execution of execution-instructions, comprising:

obtaining an execution-instruction at a processing resource;

appending a literal constant in a literal register, if a literal flag is set by examining a special register and if there is literal prefix in the execution-instruction;

executing an execution-instruction with a constant in a literal register, if a literal flag is set by examining a special register and if there is no literal prefix in the execution-instruction;

setting a literal constant flag in a status register and placing a literal constant in a literal register, if a literal flag is not set by examining a special register and if there is literal prefix in the execution-instruction;

executing an execution-instruction, if a literal flag is not set by examining a special register and if there is no literal prefix in the execution-instruction

320. The method of claim 319, wherein the method is completed within a single processing cycle.

321. The method of claim 319, wherein the execution-instruction is an execution-instruction signal for execution by a type of processing resource.

322. The method of claim 321, further comprising,

routing the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

323. The method of claim 321, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

324. The method of claim 321, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

325. The method of claim 321, wherein the execution-instruction signal includes an operation-code.

326. The method of claim 325, wherein the operation-code indicates a type of resource on which to execute.

327. The method of claim 321, wherein the execution-instruction signal includes values from a status register.

328. The method of claim 321, wherein the execution-instruction signal includes values from a priority bit status register.

329. The method of claim 321, wherein the execution-instruction signal includes a processing resource identifier.

330. The method of claim 329, wherein the processing resource identifier is an integer processing unit number.

331. The method of claim 321, wherein the execution-instruction signal includes an address.

332. The method of claim 319, wherein an other processing resource may be an delegating processing resource.

333. The method of claim 319, wherein a processing resource is an integer processing unit.

334. The method of claim 319, wherein a processing resource is a mathematical processing unit.

335. The method of claim 319, wherein a processing resource is a memory management unit.

336. The method of claim 319, wherein a processing resource is a vector processing unit.

337. The method of claim 319, wherein a processing resource is a digital signal processing unit.

338. The method of claim 319, wherein a processing resource is a graphics processing unit.

339. The method of claim 319, wherein a processing resource is an input/output controller processing unit.

340. The method of claim 319, wherein a processing resource is an execution-instruction processing cache.

341. The method of claim 319, wherein delegation occurs through an execution-instruction signal router.

342. The method of claim 341, wherein the execution-instruction signal router is a cross-point switch.

343. The method of claim 319, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

344. The method of claim 319, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

345. The method of claim 319, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

346. The method of claim 319, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

347. The method of claim 343, wherein a sleeping processing resource may be woken by a response to a request.

348. The method of claim 347, wherein the response is directed to a specific processing resource.

349. The method of claim 347, wherein the response indicates a once locked resource is unlocked.

350. The method of claim 319, wherein processing resources are communicatively disposed on a same die.

351. The method of claim 350, wherein an execution-instruction signal router is on the same die with processing resources.

352. The method of claim 319, wherein the literal constant is appended by employing a shift.

353. The method of claim 319, wherein the processing resource appends the literal constant.

354. The method of claim 319, wherein the execution-instruction is executed as an extended execution of an operation-code.

355. The method of claim 354, wherein a processing resource executes the execution-instruction.

356. The method of claim 319, wherein the status register is in the processing resource.

357. The method of claim 319, wherein the literal register is in the processing resource.

358. The method of claim 319, wherein the execution-instruction is executed as a non-extended execution of an operation-code.

359. The method of claim 358, wherein a processing resource executes the execution-instruction.

360. A method of binding instructions, comprising:

storing binding names at odd addresses;

generating a binding name interrupt, if a processing resource attempts to load an instruction register with an odd address value;

providing an operating system with a binding name interrupt;

performing a look-up of the odd address value that generated the binding name interrupt in a binding name table, which was established by linker;

replacing the odd address value in an instruction register with an even address found in the binding name table.

361. The method of claim 360, wherein the interrupt is an execution-instruction signal for execution by a type of processing resource.

362. The method of claim 361, further comprising,

routing the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

363. The method of claim 361, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

364. The method of claim 361, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

365. The method of claim 361, wherein the execution-instruction signal includes an operation-code.

366. The method of claim 365, wherein the operation-code indicates a type of resource on which to execute.

367. The method of claim 361, wherein the execution-instruction signal includes values from a status register.

368. The method of claim 361, wherein the execution-instruction signal includes values from a priority bit status register.

369. The method of claim 361, wherein the execution-instruction signal includes a processing resource identifier.

370. The method of claim 369, wherein the processing resource identifier is an integer processing unit number.

371. The method of claim 361, wherein the execution-instruction signal includes an address.

372. The method of claim 360, wherein an other processing resource may be an delegating processing resource.

373. The method of claim 360, wherein a processing resource is an integer processing unit.

374. The method of claim 360, wherein a processing resource is a mathematical processing unit.

375. The method of claim 360, wherein a processing resource is a memory management unit.

376. The method of claim 360, wherein a processing resource is a vector processing unit.

377. The method of claim 360, wherein a processing resource is a digital signal processing unit.

378. The method of claim 360, wherein a processing resource is a graphics processing unit.

379. The method of claim 360, wherein a processing resource is an input/output controller processing unit.

380. The method of claim 360, wherein a processing resource is an execution-instruction processing cache.

381. The method of claim 360, wherein delegation occurs through an execution-instruction signal router.

382. The method of claim 381, wherein the execution-instruction signal router is a cross-point switch.

383. The method of claim 360, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

384. The method of claim 360, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

385. The method of claim 360, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

386. The method of claim 360, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

387. The method of claim 383, wherein a sleeping processing resource may be woken by a response to a request.

388. The method of claim 387, wherein the response is directed to a specific processing resource.

389. The method of claim 387, wherein the response indicates a once locked resource is unlocked.

390. The method of claim 360, wherein processing resources are communicatively disposed on a same die.

391. The method of claim 390, wherein an execution-instruction signal router is on the same die with processing resources.

392. The method of claim 360, wherein the binding names are stored by a linker.

393. The method of claim 360, further comprising,

replacing the odd address value stored in an instruction stack with an instruction pointer at the even address, if late binding is being employed.

394. The method of claim 360, further comprising,

replacing the odd address value stored in an instruction stack with an instruction pointer at the even address and replacing the odd address value in the binding name table with the even address, if dynamic binding is being employed.

395. A method of addressing registers, comprising:

determining if an execution-instruction operation-code provides additional register addressing information;

setting cycle flags in a status register with the additional register addressing information;

examining an execution-instruction for a register address;

addressing a register specified by the register address and cycle flags.

396. The method of claim 395, wherein the method is completed within a single processing cycle.

397. The method of claim 395, wherein the execution-instruction is an execution-instruction signal for execution by a type of processing resource.

398. The method of claim 397, further comprising,

routing the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

399. The method of claim 397, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

400. The method of claim 397, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

401. The method of claim 397, wherein the execution-instruction signal includes an operation-code.

402. The method of claim 401, wherein the operation-code indicates a type of resource on which to execute.

403. The method of claim 397, wherein the execution-instruction signal includes values from a status register.

404. The method of claim 397, wherein the execution-instruction signal includes values from a priority bit status register.

405. The method of claim 397, wherein the execution-instruction signal includes a processing resource identifier.

406. The method of claim 405, wherein the processing resource identifier is an integer processing unit number.

407. The method of claim 397, wherein the execution-instruction signal includes an address.

408. The method of claim 395, wherein an other processing resource may be an delegating processing resource.

409. The method of claim 395, wherein a processing resource is an integer processing unit.

410. The method of claim 395, wherein a processing resource is a mathematical processing unit.

411. The method of claim 395, wherein a processing resource is a memory management unit.

412. The method of claim 395, wherein a processing resource is a vector processing unit.

413. The method of claim 395, wherein a processing resource is a digital signal processing unit.

414. The method of claim 395, wherein a processing resource is a graphics processing unit.

415. The method of claim 395, wherein a processing resource is an input/output controller processing unit.

416. The method of claim 395, wherein a processing resource is an execution-instruction processing cache.

417. The method of claim 395, wherein delegation occurs through an execution-instruction signal router.

418. The method of claim 417, wherein the execution-instruction signal router is a cross-point switch.

419. The method of claim 395, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

420. The method of claim 395, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

421. The method of claim 395, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

422. The method of claim 395, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

423. The method of claim 419, wherein a sleeping processing resource may be woken by a response to a request.

424. The method of claim 423, wherein the response is directed to a specific processing resource.

425. The method of claim 423, wherein the response indicates a once locked resource is unlocked.

426. The method of claim 395, wherein processing resources are communicatively disposed on a same die.

427. The method of claim 426, wherein an execution-instruction signal router is on the same die with processing resources.

428. The method of claim 395, wherein specific operation-codes address extended register sets without requiring dedicated bits set aside for register addressing in the execution-instruction.

429. The method of claim 395, wherein the register address comprises 3 bits of the execution-instruction.

430. The method of claim 395, wherein a value of a cycle flag indicates that an operation-code is a multi-cycle operation.

431. A method of sharing memory, comprising:

obtaining a request execution-instruction from a delegate processing resource at an instruction processing cache, wherein the request includes a target memory address;

determining if the target address is locked based on a lock variable at the target memory address;

determining what type of lock is provided in the request;

requesting that a requesting processing resource sleep until it is unlocked, if the target memory address is locked.

432. The method of claim 431, wherein the method is completed within a single processing cycle.

433. The method of claim 431, wherein the request execution-instruction is an execution-instruction signal for execution by a type of processing resource.

434. The method of claim 433, further comprising,

routing the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

435. The method of claim 433, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

436. The method of claim 433, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

437. The method of claim 433, wherein the execution-instruction signal includes an operation-code.

438. The method of claim 437, wherein the operation-code indicates a type of resource on which to execute.

439. The method of claim 433, wherein the execution-instruction signal includes values from a status register.

440. The method of claim 433, wherein the execution-instruction signal includes values from a priority bit status register.

441. The method of claim 433, wherein the execution-instruction signal includes a processing resource identifier.

442. The method of claim 441, wherein the processing resource identifier is an integer processing unit number.

443. The method of claim 433, wherein the execution-instruction signal includes an address.

444. The method of claim 431, wherein an other processing resource may be an delegating processing resource.

445. The method of claim 431, wherein a processing resource is an integer processing unit.

446. The method of claim 431, wherein a processing resource is a mathematical processing unit.

447. The method of claim 431, wherein a processing resource is a memory management unit.

448. The method of claim 431, wherein a processing resource is a vector processing unit.

449. The method of claim 431, wherein a processing resource is a digital signal processing unit.

450. The method of claim 431, wherein a processing resource is a graphics processing unit.

451. The method of claim 431, wherein a processing resource is an input/output controller processing unit.

452. The method of claim 431, wherein a processing resource is an execution-instruction processing cache.

453. The method of claim 431, wherein delegation occurs through an execution-instruction signal router.

454. The method of claim 453, wherein the execution-instruction signal router is a cross-point switch.

455. The method of claim 431, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

456. The method of claim 431, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

457. The method of claim 431, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

458. The method of claim 431, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

459. The method of claim 455, wherein a sleeping processing resource may be woken by a response to a request.

460. The method of claim 459, wherein the response is directed to a specific processing resource.

461. The method of claim 459, wherein the response indicates a once locked resource is unlocked.

462. The method of claim 431, wherein processing resources are communicatively disposed on a same die.

463. The method of claim 462, wherein an execution-instruction signal router is on the same die with processing resources.

464. The method of claim 431, wherein the determination is made by the instruction processing cache's logic facilities.

465. The method of claim 431, wherein the requesting processing resource is unlocked when an unlock instruction is received that specifies the same target memory address.

466. The method of claim 431, wherein the target memory address is locked for read and write operations.

467. The method of claim 431, wherein the target memory address is locked for read only operations and the request specifies a read and write operation lock.

468. A method of sharing memory, comprising:

obtaining a response to an execution-instruction from a delegate processing resource, wherein the response includes a target address at a processing resource;

determining if a target address is locked based on a lock variable at the target memory address;

determining value types by using a hash function;

updating value types in a primary cache memory of a processing resource to a secondary cache memory for each value type, if each value type in a primary cache memory of a processing resource has not been updated to a secondary cache memory;

469. The method of claim 468, wherein the method is completed within a single processing cycle. \

470. The method of claim 468, wherein the response is an execution-instruction signal for execution by a type of processing resource.

471. The method of claim 470, further comprising,

routing the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

472. The method of claim 470, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

473. The method of claim 470, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

474. The method of claim 470, wherein the execution-instruction signal includes an operation-code.

475. The method of claim 474, wherein the operation-code indicates a type of resource on which to execute.

476. The method of claim 470, wherein the execution-instruction signal includes values from a status register.

477. The method of claim 470, wherein the execution-instruction signal includes values from a priority bit status register.

478. The method of claim 470, wherein the execution-instruction signal includes a processing resource identifier.

479. The method of claim 478, wherein the processing resource identifier is an integer processing unit number.

480. The method of claim 470, wherein the execution-instruction signal includes an address.

481. The method of claim 468, wherein an other processing resource may be an delegating processing resource.

482. The method of claim 468, wherein a processing resource is an integer processing unit.

483. The method of claim 468, wherein a processing resource is a mathematical processing unit.

484. The method of claim 468, wherein a processing resource is a memory management unit.

485. The method of claim 468, wherein a processing resource is a vector processing unit.

486. The method of claim 468, wherein a processing resource is a digital signal processing unit.

487. The method of claim 468, wherein a processing resource is a graphics processing unit.

488. The method of claim 468, wherein a processing resource is an input/output controller processing unit.

489. The method of claim 468, wherein a processing resource is an execution-instruction processing cache.

490. The method of claim 468, wherein delegation occurs through an execution-instruction signal router.

491. The method of claim 490, wherein the execution-instruction signal router is a cross-point switch.

492. The method of claim 468, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

493. The method of claim 468, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

494. The method of claim 468, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

495. The method of claim 468, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

496. The method of claim 492, wherein a sleeping processing resource may be woken by a response to a request.

497. The method of claim 496, wherein the response is directed to a specific processing resource.

498. The method of claim 496, wherein the response indicates a once locked resource is unlocked.

499. The method of claim 468, wherein processing resources are communicatively disposed on a same die.

500. The method of claim 499, wherein an execution-instruction signal router is on the same die with processing resources.

501. The method of claim 468, wherein the response includes a target processing resource identifier.

502. The method of claim 468, wherein the determination is made by a processing resource.

503. The method of claim 468, wherein the value type is a global value.

504. The method of claim 468, wherein the secondary cache memory is a Level 2 cache memory.

505. The method of claim 468, wherein a primary cache memory is a Level 1 cache memory.

506. The method of claim 468, further comprising,

marking a cache line of the updated value type as free. 507. The method of claim 468, further comprising,

updating lock variables in secondary cache memory for each updated value type.

507. The method of claim 507, wherein the lock variable updating is performed by the secondary cache memory, which is a processing cache memory.

508. A method of sharing memory, comprising:

obtaining a storage request including an event variable address and execution-instruction for a processing resource to sleep;

determining if an event variable is set;

providing a reply to a processing resource bus with the event variable address and event value, which will wake an appropriate processing resource from sleep, if the event value is set not to sleep;

setting a sleep until event value for the event variable, if the event value is set to sleep.

509. A method of sharing memory, comprising:

obtaining a storage request including an event variable address and execution instruction for a processing resource to sleep;

determining if an event variable is set;

providing a reply to a processing resource bus with the event variable address and event value, which will wake an appropriate processing resource from sleep, if the event value is set not to sleep;

setting a sleep until event value for the event variable, if the event value is set to sleep.

510. The method of claim 509, wherein the method is completed within a single processing cycle.

511. The method of claim 509, wherein the storage request is an execution-instruction signal for execution by a type of processing resource.

512. The method of claim 511, further comprising,

routing the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

513. The method of claim 511, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

514. The method of claim 511, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

515. The method of claim 511, wherein the execution-instruction signal includes an operation-code.

516. The method of claim 515, wherein the operation-code indicates a type of resource on which to execute.

517. The method of claim 511, wherein the execution-instruction signal includes values from a status register.

518. The method of claim 511, wherein the execution-instruction signal includes values from a priority bit status register.

519. The method of claim 511, wherein the execution-instruction signal includes a processing resource identifier.

520. The method of claim 519, wherein the processing resource identifier is an integer processing unit number.

521. The method of claim 511, wherein the execution-instruction signal includes an address.

522. The method of claim 509, wherein an other processing resource may be an delegating processing resource.

523. The method of claim 509, wherein a processing resource is an integer processing unit.

524. The method of claim 509, wherein a processing resource is a mathematical processing unit.

525. The method of claim 509, wherein a processing resource is a memory management unit.

526. The method of claim 509, wherein a processing resource is a vector processing unit.

527. The method of claim 509, wherein a processing resource is a digital signal processing unit.

528. The method of claim 509, wherein a processing resource is a graphics processing unit.

529. The method of claim 509, wherein a processing resource is an input/output controller processing unit.

530. The method of claim 509, wherein a processing resource is an execution-instruction processing cache.

531. The method of claim 509, wherein delegation occurs through an execution-instruction signal router.

532. The method of claim 531, wherein the execution-instruction signal router is a cross-point switch.

533. The method of claim 509, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

534. The method of claim 509, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

535. The method of claim 509, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

536. The method of claim 509, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

537. The method of claim 533, wherein a sleeping processing resource may be woken by a response to a request.

538. The method of claim 537, wherein the response is directed to a specific processing resource.

539. The method of claim 537, wherein the response indicates a once locked resource is unlocked.

540. The method of claim 509, wherein processing resources are communicatively disposed on a same die.

541. The method of claim 540, wherein an execution-instruction signal router is on the same die with processing resources.

542. The method of claim 509, wherein the determination is made at a processing cache.

543. The method of claim 509, wherein the event variable is set at the processing cache.

544. The method of claim 509, wherein the event variable is set in a register.

545. The method of claim 509, wherein the event variable is set in a target address in a processing cache.

546. A method of sharing memory, comprising:

obtaining a storage request including an event address at a processing cache;

combining a parameter with an event variable through in a processing cache;

providing a reply to a processing resource bus from the combination with the event address, which will wake an appropriate processing resource from sleep that is waiting on an event, if the event value is set not to sleep.

547. The method of claim 546, wherein the method is completed within a single processing cycle.

548. The method of claim 546, wherein the storage request is an execution-instruction signal for execution by a type of processing resource.

549. The method of claim 548, further comprising,

routing the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

550. The method of claim 548, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

551. The method of claim 548, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

552. The method of claim 548, wherein the execution-instruction signal includes an operation-code.

553. The method of claim 552, wherein the operation-code indicates a type of resource on which to execute.

554. The method of claim 548, wherein the execution-instruction signal includes values from a status register.

555. The method of claim 548, wherein the execution-instruction signal includes values from a priority bit status register.

556. The method of claim 548, wherein the execution-instruction signal includes a processing resource identifier.

557. The method of claim 556, wherein the processing resource identifier is an integer processing unit number.

558. The method of claim 548, wherein the execution-instruction signal includes an address.

559. The method of claim 546, wherein an other processing resource may be an delegating processing resource.

560. The method of claim 546, wherein a processing resource is an integer processing unit.

561. The method of claim 546, wherein a processing resource is a mathematical processing unit.

562. The method of claim 546, wherein a processing resource is a memory management unit.

563. The method of claim 546, wherein a processing resource is a vector processing unit.

564. The method of claim 546, wherein a processing resource is a digital signal processing unit.

565. The method of claim 546, wherein a processing resource is a graphics processing unit.

566. The method of claim 546, wherein a processing resource is an input/output controller processing unit.

567. The method of claim 546, wherein a processing resource is an execution-instruction processing cache.

568. The method of claim 546, wherein delegation occurs through an execution-instruction signal router.

569. The method of claim 568, wherein the execution-instruction signal router is a cross-point switch.

570. The method of claim 546, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

571. The method of claim 546, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

572. The method of claim 546, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

573. The method of claim 546, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

574. The method of claim 570, wherein a sleeping processing resource may be woken by a response to a request.

575. The method of claim 574, wherein the response is directed to a specific processing resource.

576. The method of claim 574, wherein the response indicates a once locked resource is unlocked.

577. The method of claim 546, wherein processing resources are communicatively disposed on a same die.

578. The method of claim 577, wherein an execution-instruction signal router is on the same die with processing resources.

579. The method of claim 546, wherein the parameter is a bit-mask.

580. The method of claim 546, wherein the combination is a logical-OR.

581. A method of sharing memory, comprising:

obtaining a storage request including a semaphore address at a processing cache;

setting a semaphore variable in a processing cache at the semaphore address;

providing an operating system trap-call to wait on a processing queue, if the storage request includes a wait-on-semaphore instruction;

providing an operating system trap-call to signal on a processing queue, if the storage request includes a signal-on-semaphore instruction;

582. The method of claim 581, wherein the method is completed within a single processing cycle.

583. The method of claim 581, wherein the storage request is an execution-instruction signal for execution by a type of processing resource.

584. The method of claim 583, further comprising,

routing the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

585. The method of claim 583, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

586. The method of claim 583, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

587. The method of claim 583, wherein the execution-instruction signal includes an operation-code.

588. The method of claim 587, wherein the operation-code indicates a type of resource on which to execute.

589. The method of claim 583, wherein the execution-instruction signal includes values from a status register.

590. The method of claim 583, wherein the execution-instruction signal includes values from a priority bit status register.

591. The method of claim 583, wherein the execution-instruction signal includes a processing resource identifier.

592. The method of claim 591, wherein the processing resource identifier is an integer processing unit number.

593. The method of claim 583, wherein the execution-instruction signal includes an address.

594. The method of claim 581, wherein an other processing resource may be an delegating processing resource.

595. The method of claim 581, wherein a processing resource is an integer processing unit.

596. The method of claim 581, wherein a processing resource is a mathematical processing unit.

597. The method of claim 581, wherein a processing resource is a memory management unit.

598. The method of claim 581, wherein a processing resource is a vector processing unit.

599. The method of claim 581, wherein a processing resource is a digital signal processing unit.

600. The method of claim 581, wherein a processing resource is a graphics processing unit.

601. The method of claim 581, wherein a processing resource is an input/output controller processing unit.

602. The method of claim 581, wherein a processing resource is an execution-instruction processing cache.

603. The method of claim 581, wherein delegation occurs through an execution-instruction signal router.

604. The method of claim 603, wherein the execution-instruction signal router is a cross-point switch.

605. The method of claim 581, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

606. The method of claim 581, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

607. The method of claim 581, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

608. The method of claim 581, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

609. The method of claim 605, wherein a sleeping processing resource may be woken by a response to a request.

610. The method of claim 609, wherein the response is directed to a specific processing resource.

611. The method of claim 609, wherein the response indicates a once locked resource is unlocked.

612. The method of claim 581, wherein processing resources are communicatively disposed on a same die.

613. The method of claim 612, wherein an execution-instruction signal router is on the same die with processing resources.

614. The method of claim 581, wherein the trap-call causes rescheduling so that other processes can run on a particular processing resource.

615. The method of claim 581, wherein other processes include threads.

616. The method of claim 581, wherein other processes include threads.

617. A system of register addressing, comprising:

means to obtain a register bank address;

means to decode an operation-code;

means to set a cycle flag in a status register based on the decoding of the operation-code;.

means to access a register addressed by the cycle flag within the addressed register bank.

618. The system of claim 617, further comprising,

means to employ a set number of bits in an execution-instruction to establish the number of registers within a register bank.

619. The system of claim 617, further comprising,

means to employ a set number of bits in an execution-instruction to establish the number of register banks.

620. A system of setting intra-processor processing resources to sleep, comprising:

means to set processing resources that issue requests to delegating processing resources to sleep until a response is provided;

means to set processing resources waiting on shared and locked memory being accessed by other processing resources to sleep until the memory is unlocked.

621. A system of execution-instruction delegation between processing resources, comprising:

means to share memory between processing resources

wherein the memory itself includes instruction execution logic;

wherein the processing resources are communicatively accessible through an instruction execution router;

wherein the processing resources are on the same die;

means to delegate execution-instructions from originating processing resources to other processing resources.

622. The system of claim 621, wherein the system is completed within a single processing cycle.

623. The system of claim 621, wherein the other processing resource may be the originating processing resource.

624. The system of claim 621, wherein a processing resource is an integer processing unit.

625. The system of claim 621, wherein a processing resource is a mathematical processing unit.

626. The system of claim 621, wherein a processing resource is a memory management unit.

627. The system of claim 621, wherein a processing resource is a vector processing unit.

628. The system of claim 621, wherein a processing resource is a digital signal processing unit.

629. The system of claim 621, wherein a processing resource is a graphics processing unit.

630. The system of claim 621, wherein a processing resource is an input/output controller processing unit.

631. The system of claim 621, wherein a processing resource is an execution-instruction processing cache.

632. The system of claim 621, wherein delegation occurs through an execution-instruction signal router.

633. The system of claim 632, wherein the execution-instruction signal router is a cross-point switch.

634. The system of claim 621, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

635. The system of claim 621, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

636. The system of claim 621, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

637. The system of claim 621, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

638. The system of claim 634, wherein a sleeping processing resource may be woken by a response to a request.

639. The system of claim 638, wherein the response is directed to a specific processing resource.

640. The system of claim 638, wherein the response indicates a once locked resource is unlocked.

641. The system of claim 621, wherein processing resources are communicatively disposed on a same die.

642. The system of claim 641, wherein an execution-instruction signal router is on the same die with processing resources.

643. A system of execution-instruction delegation between processing resources, comprising:

means to obtain an execution instruction, wherein the execution instruction is obtained at a processing resource;

means to determine whether an operation-code within the execution instruction should be delegated to an other processing resource;

means to execute the execution instruction, if the operation-code within the execution instruction should not be delegated to an other processing resource;

means to route the execution instruction to an other processing resource, if the operation-code within the execution instruction is for an other processing resource.

644. The system of claim 643, wherein the system is completed within a single processing cycle.

645. The system of claim 643, wherein the routed execution instruction is a request for execution by a type of processing resource.

646. The system of claim 645, wherein the request includes an operation-code.

647. The system of claim 646, wherein the operation-code indicates a type of resource on which to execute.

648. The system of claim 645, wherein the request includes values from a status register.

649. The system of claim 645, wherein the request includes values from a priority bit status register.

650. The system of claim 645, wherein the request includes a processing resource identifier.

651. The system of claim 650, wherein the processing resource identifier is an integer processing unit number.

652. The system of claim 645, wherein the request is obtained by an execution-instruction signal router.

653. The system of claim 645, wherein the request includes an address.

654. The system of claim 643, wherein the operation-code indicates a type of resource on which to execute.

655. The system of claim 643, wherein the other processing resource may be the originating processing resource.

656. The system of claim 643, wherein a processing resource is an integer processing unit.

657. The system of claim 643, wherein a processing resource is a mathematical processing unit.

658. The system of claim 643, wherein a processing resource is a memory management unit.

659. The system of claim 643, wherein a processing resource is a vector processing unit.

660. The system of claim 643, wherein a processing resource is a digital signal processing unit.

661. The system of claim 643, wherein a processing resource is a graphics processing unit.

662. The system of claim 643, wherein a processing resource is an input/output controller processing unit.

663. The system of claim 643, wherein a processing resource is an execution-instruction processing cache.

664. The system of claim 643, wherein routing occurs through an execution-instruction signal router.

665. The system of claim 664, wherein the execution-instruction signal router is a cross-point switch.

666. The system of claim 643, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

667. The system of claim 643, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

668. The system of claim 643, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

669. The system of claim 643, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

670. The system of claim 643, wherein processing resources are communicatively disposed on a same die.

671. The system of claim 670, wherein an execution-instruction signal router is on the same die with processing resources.

672. A system of execution-instruction delegation between processing resources, comprising:

means to obtain an request signal, wherein the request signal is from a requesting processing resource;

means to determine a priority dead-lock-avoidance value, wherein the priority dead-lock value is used to select among multiple requests with equal priority;

means to examine a request priority value for each submitted request;

means to select a request with a highest priority value, if more than one processing resource requests a same target resource;

means to provide a selected request to a target processing resource.

673. The system of claim 672, wherein the system is completed within a single processing cycle.

674. The system of claim 672, wherein the request signal is a request for execution by a type of processing resource.

675. The system of claim 674, further comprising,

means to route the request to an other processing resource, if an operation-code within the request is for an other processing resource.

676. The system of claim 674, wherein the request is obtained at an execution-instruction signal router.

677. The system of claim 674, wherein the execution-instruction signal is provided by a delegating processing resource via an execution-instruction signal router.

678. The system of claim 674, wherein the request includes an operation-code.

679. The system of claim 678, wherein the operation-code indicates a type of resource on which to execute.

680. The system of claim 674, wherein the request includes values from a status register.

681. The system of claim 674, wherein the request includes values from a priority bit status register.

682. The system of claim 674, wherein the request includes a processing resource identifier.

683. The system of claim 682, wherein the processing resource identifier is an integer processing unit number.

684. The system of claim 674, wherein the request includes an address.

685. The system of claim 672, wherein the other processing resource may be the originating processing resource.

686. The system of claim 672, wherein a processing resource is an integer processing unit.

687. The system of claim 672, wherein a processing resource is a mathematical processing unit.

688. The system of claim 672, wherein a processing resource is a memory management unit.

689. The system of claim 672, wherein a processing resource is a vector processing unit.

690. The system of claim 672, wherein a processing resource is a digital signal processing unit.

691. The system of claim 672, wherein a processing resource is a graphics processing unit.

692. The system of claim 672, wherein a processing resource is an input/output controller processing unit.

693. The system of claim 672, wherein a processing resource is an execution-instruction processing cache.

694. The system of claim 672, wherein delegation occurs through an execution-instruction signal router.

695. The system of claim 694, wherein the execution-instruction signal router is a cross-point switch.

696. The system of claim 672, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

697. The system of claim 672, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

698. The system of claim 672, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

699. The system of claim 672, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

700. The system of claim 696, wherein a sleeping processing resource may be woken by a response to a request.

701. The system of claim 700, wherein the response is directed to a specific processing resource.

702. The system of claim 700, wherein the response indicates a once locked resource is unlocked.

703. The system of claim 672, wherein processing resources are communicatively disposed on a same die.

704. The system of claim 703, wherein an execution-instruction signal router is on the same die with processing resources.

705. The system of claim 672, wherein the priority dead-lock-avoidance value is iterated.

706. The system of claim 672, further comprising,

means to select a request, if not more than one processing resource requests a same target resource.

707. The system of claim 672, further comprising,

means to select the request with a highest priority value, if not more than one request has a same highest priority value;

means to select a request based on the priority dead-lock-avoidance value, if more than one request has a same highest priority value.

708. The system of claim 672, further comprising,

means to provide a request granted acknowledgement to the requesting processing resource whose request was selected.

709. The system of claim 672, further comprising,

means to clear the operation-code in the requesting processing resource whose request was selected.

710. A system of execution-instruction delegation between processing resources, comprising:

means to process a request execution-instruction, wherein the request execution-instruction includes a requesting processing resource identifier;

means to prepare a response into a result register, wherein the response includes the requesting processing resource identifier;

means to present the response to all processing resources.

711. The system of claim 710, wherein the system is completed within a single processing cycle.

712. The system of claim 710, wherein the request execution-instruction is a request for execution by a type of processing resource.

713. The system of claim 712, further comprising,

means to route the request to an other processing resource, if an operation-code within the request is for an other processing resource.

714. The system of claim 712, wherein the request is obtained at an execution-instruction signal router.

715. The system of claim 712, wherein the execution-instruction signal is provided by a delegating processing resource via an execution-instruction signal router.

716. The system of claim 712, wherein the request includes an operation-code.

717. The system of claim 716, wherein the operation-code indicates a type of resource on which to execute.

718. The system of claim 712, wherein the request includes values from a status register.

719. The system of claim 712, wherein the request includes values from a priority bit status register.

720. The system of claim 712, wherein the request includes a processing resource identifier.

721. The system of claim 720, wherein the processing resource identifier is an integer processing unit number.

722. The system of claim 712, wherein the request includes an address.

723. The system of claim 710, wherein the other processing resource may be the originating processing resource.

724. The system of claim 710, wherein a processing resource is an integer processing unit.

725. The system of claim 710, wherein a processing resource is a mathematical processing unit.

726. The system of claim 710, wherein a processing resource is a memory management unit.

727. The system of claim 710, wherein a processing resource is a vector processing unit.

728. The system of claim 710, wherein a processing resource is a digital signal processing unit.

729. The system of claim 710, wherein a processing resource is a graphics processing unit.

730. The system of claim 710, wherein a processing resource is an input/output controller processing unit.

731. The system of claim 710, wherein a processing resource is an execution-instruction processing cache.

732. The system of claim 710, wherein delegation occurs through an execution-instruction signal router.

733. The system of claim 732, wherein the execution-instruction signal router is a cross-point switch.

734. The system of claim 710, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

735. The system of claim 710, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

736. The system of claim 710, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

737. The system of claim 710, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

738. The system of claim 734, wherein a sleeping processing resource may be woken by a response to a request.

739. The system of claim 738, wherein the response is directed to a specific processing resource.

740. The system of claim 738, wherein the response indicates a once locked resource is unlocked.

741. The system of claim 710, wherein processing resources are communicatively disposed on a same die.

742. The system of claim 741, wherein an execution-instruction signal router is on the same die with processing resources.

743. A system of execution-instruction delegation between processing resources, comprising:

means to obtain a result execution-instruction from a delegate processing resource, wherein the result includes a requesting processing resource identifier;

means to wake the requesting processing resource, if the requesting processing resource identifier identifies the instant processing resource;

means to wake a processing resource, if a processing resource is waiting to be unlocked and if the obtained result includes an unlock flag and if an address in the result matches a locked address of an instant processing resource.

744. The system of claim 743, wherein the system is completed within a single processing cycle.

745. The system of claim 743, wherein the result execution-instruction is an execution-instruction signal for execution by a type of processing resource.

746. The system of claim 745, further comprising,

means to route the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

747. The system of claim 745, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

748. The system of claim 745, wherein the execution-instruction signal is provided by a delegating processing resource via an execution-instruction signal router.

749. The system of claim 745, wherein the execution-instruction signal includes an operation-code.

750. The system of claim 749, wherein the operation-code indicates a type of resource on which to execute.

751. The system of claim 745, wherein the execution-instruction signal includes values from a status register.

752. The system of claim 745, wherein the execution-instruction signal includes values from a priority bit status register.

753. The system of claim 745, wherein the execution-instruction signal includes a processing resource identifier.

754. The system of claim 753, wherein the processing resource identifier is an integer processing unit number.

755. The system of claim 745, wherein the execution-instruction signal includes an address.

756. The system of claim 743, wherein an other processing resource may be an delegating processing resource.

757. The system of claim 743, wherein a processing resource is an integer processing unit.

758. The system of claim 743, wherein a processing resource is a mathematical processing unit.

759. The system of claim 743, wherein a processing resource is a memory management unit.

760. The system of claim 743, wherein a processing resource is a vector processing unit.

761. The system of claim 743, wherein a processing resource is a digital signal processing unit.

762. The system of claim 743, wherein a processing resource is a graphics processing unit.

763. The system of claim 743, wherein a processing resource is an input/output controller processing unit.

764. The system of claim 743, wherein a processing resource is an execution-instruction processing cache.

765. The system of claim 743, wherein delegation occurs through an execution-instruction signal router.

766. The system of claim 765, wherein the execution-instruction signal router is a cross-point switch.

767. The system of claim 743, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

768. The system of claim 743, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

769. The system of claim 743, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

770. The system of claim 743, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

771. The system of claim 767, wherein a sleeping processing resource may be woken by a response to a request.

772. The system of claim 771, wherein the response is directed to a specific processing resource.

773. The system of claim 771, wherein the response indicates a once locked resource is unlocked.

774. The system of claim 743, wherein processing resources are communicatively disposed on a same die.

775. The system of claim 774, wherein an execution-instruction signal router is on the same die with processing resources.

776. The system of claim 743, further comprising,

means to execute a next execution-instruction, if the requesting processing resource identifier identifies the instant processing resource;

means to execute an execution-instruction that failed to execute, if a processing resource is waiting to be unlocked and if the obtained result includes an unlock flag and if an address in the result matches a locked address of an instant processing resource.

777. A system of memory access optimization, comprising:

means to obtain a storage-access request from a delegated processing resource, wherein the request includes a target memory address;

means to determine the target memory address from the request;

means to compare the target memory address with register values, wherein register values are used to establish a data type of the target memory address for subsequent storage in an apportioned region of cache memory;

means to obtain information from the target memory address;

means to store the information in an apportioned region of cache memory.

778. The system of claim 777, wherein the system is completed within a single processing cycle.

779. The system of claim 777, wherein the storage-access request is an execution-instruction signal for execution by a type of processing resource.

780. The system of claim 779, further comprising,

means to route the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

781. The system of claim 779, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

782. The system of claim 779, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

783. The system of claim 779, wherein the execution-instruction signal includes an operation-code.

784. The system of claim 783, wherein the operation-code indicates a type of resource on which to execute.

785. The system of claim 779, wherein the execution-instruction signal includes values from a status register.

786. The system of claim 779, wherein the execution-instruction signal includes values from a priority bit status register.

787. The system of claim 779, wherein the execution-instruction signal includes a processing resource identifier.

788. The system of claim 787, wherein the processing resource identifier is an integer processing unit number.

789. The system of claim 779, wherein the execution-instruction signal includes an address.

790. The system of claim 777, wherein an other processing resource may be an delegating processing resource.

791. The system of claim 777, wherein a processing resource is an integer processing unit.

792. The system of claim 777, wherein a processing resource is a mathematical processing unit.

793. The system of claim 777, wherein a processing resource is a memory management unit.

794. The system of claim 777, wherein a processing resource is a vector processing unit.

795. The system of claim 777, wherein a processing resource is a digital signal processing unit.

796. The system of claim 777, wherein a processing resource is a graphics processing unit.

797. The system of claim 777, wherein a processing resource is an input/output controller processing unit.

798. The system of claim 777, wherein a processing resource is an execution-instruction processing cache.

799. The system of claim 777, wherein delegation occurs through an execution-instruction signal router.

800. The system of claim 799, wherein the execution-instruction signal router is a cross-point switch.

801. The system of claim 777, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

802. The system of claim 777, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

803. The system of claim 777, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

804. The system of claim 777, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

805. The system of claim 801, wherein a sleeping processing resource may be woken by a response to a request.

806. The system of claim 805, wherein the response is directed to a specific processing resource.

807. The system of claim 805, wherein the response indicates a once locked resource is unlocked.

808. The system of claim 777, wherein processing resources are communicatively disposed on a same die.

809. The system of claim 808, wherein an execution-instruction signal router is on the same die with processing resources.

810. The system of claim 777, wherein the storage-access request is obtained at a processing resource.

811. The system of claim 777, wherein information at the target memory address is designated to be local data if the target memory address is within an address range established by stack base and end register values.

812. The system of claim 777, wherein information at the target memory address is designated to be global data if the target memory address is outside an address range established by stack base and end register values.

813. The system of claim 777, wherein an apportionment is designated for each type of data for optimized access.

814. The system of claim 777, wherein the apportioned region is a hash region.

815. The system of claim 814, wherein a hash region is designated for local data.

816. The system of claim 814, wherein a hash region is designated for global data.

817. The system of claim 814, wherein the apportionment for a local data region is greater than a region for global data.

818. A system of reduced size execution of execution-instructions, comprising:

means to obtain a an execution-instruction at a processing resource;

means to append a literal constant in a literal register, if a literal flag is set by examining a special register and if there is literal prefix in the execution-instruction;

means to execute an execution-instruction with a constant in a literal register, if a literal flag is set by examining a special register and if there is no literal prefix in the execution-instruction;

means to set a literal constant flag in a status register and placing a literal constant in a literal register, if a literal flag is not set by examining a special register and if there is literal prefix in the execution-instruction;

means to execute an execution-instruction, if a literal flag is not set by examining a special register and if there is no literal prefix in the execution-instruction.

819. The system of claim 818, wherein the system is completed within a single processing cycle.

820. The system of claim 818, wherein the execution-instruction is an execution-instruction signal for execution by a type of processing resource.

821. The system of claim 820, further comprising,

means to route the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

822. The system of claim 820, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

823. The system of claim 820, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

824. The system of claim 820, wherein the execution-instruction signal includes an operation-code.

825. The system of claim 824, wherein the operation-code indicates a type of resource on which to execute.

826. The system of claim 820, wherein the execution-instruction signal includes values from a status register.

827. The system of claim 820, wherein the execution-instruction signal includes values from a priority bit status register.

828. The system of claim 820, wherein the execution-instruction signal includes a processing resource identifier.

829. The system of claim 828, wherein the processing resource identifier is an integer processing unit number.

830. The system of claim 820, wherein the execution-instruction signal includes an address.

831. The system of claim 818, wherein an other processing resource may be an delegating processing resource.

832. The system of claim 818, wherein a processing resource is an integer processing unit.

833. The system of claim 818, wherein a processing resource is a mathematical processing unit.

834. The system of claim 818, wherein a processing resource is a memory management unit.

835. The system of claim 818, wherein a processing resource is a vector processing unit.

836. The system of claim 818, wherein a processing resource is a digital signal processing unit.

837. The system of claim 818, wherein a processing resource is a graphics processing unit.

838. The system of claim 818, wherein a processing resource is an input/output controller processing unit.

839. The system of claim 818, wherein a processing resource is an execution-instruction processing cache.

840. The system of claim 818, wherein delegation occurs through an execution-instruction signal router.

841. The system of claim 840, wherein the execution-instruction signal router is a cross-point switch.

842. The system of claim 818, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

843. The system of claim 818, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

844. The system of claim 818, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

845. The system of claim 818, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

846. The system of claim 842, wherein a sleeping processing resource may be woken by a response to a request.

847. The system of claim 846, wherein the response is directed to a specific processing resource.

848. The system of claim 846, wherein the response indicates a once locked resource is unlocked.

849. The system of claim 818, wherein processing resources are communicatively disposed on a same die.

850. The system of claim 849, wherein an execution-instruction signal router is on the same die with processing resources.

851. The system of claim 818, wherein the literal constant is appended by employing a shift.

852. The system of claim 818, wherein the processing resource appends the literal constant.

853. The system of claim 818, wherein the execution-instruction is executed as an extended execution of an operation-code.

854. The system of claim 853, wherein a processing resource executes the execution-instruction.

855. The system of claim 818, wherein the status register is in the processing resource.

856. The system of claim 818, wherein the literal register is in the processing resource.

857. The system of claim 818, wherein the execution-instruction is executed as a non-extended execution of an operation-code.

858. The system of claim 857, wherein a processing resource executes the execution-instruction.

859. A system of binding instructions, comprising:

means to store binding names at odd addresses;

means to generate a binding name interrupt, if a processing resource attempts to load an instruction register with an odd address value;

means to provide an operating system with a binding name interrupt;

means to perform a look-up of the odd address value that generated the binding name interrupt in a binding name table, which was established by linker;

means to replace the odd address value in an instruction register with an even address found in the binding name table.

860. The system of claim 859, wherein the interrupt is an execution-instruction signal for execution by a type of processing resource.

861. The system of claim 860, further comprising,

means to route the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

862. The system of claim 860, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

863. The system of claim 860, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

864. The system of claim 860, wherein the execution-instruction signal includes an operation-code.

865. The system of claim 864, wherein the operation-code indicates a type of resource on which to execute.

866. The system of claim 860, wherein the execution-instruction signal includes values from a status register.

867. The system of claim 860, wherein the execution-instruction signal includes values from a priority bit status register.

868. The system of claim 860, wherein the execution-instruction signal includes a processing resource identifier.

869. The system of claim 868, wherein the processing resource identifier is an integer processing unit number.

870. The system of claim 860, wherein the execution-instruction signal includes an address.

871. The system of claim 859, wherein an other processing resource may be an delegating processing resource.

872. The system of claim 859, wherein a processing resource is an integer processing unit.

873. The system of claim 859, wherein a processing resource is a mathematical processing unit.

874. The system of claim 859, wherein a processing resource is a memory management unit.

875. The system of claim 859, wherein a processing resource is a vector processing unit.

876. The system of claim 859, wherein a processing resource is a digital signal processing unit.

877. The system of claim 859, wherein a processing resource is a graphics processing unit.

878. The system of claim 859, wherein a processing resource is an input/output controller processing unit.

879. The system of claim 859, wherein a processing resource is an execution-instruction processing cache.

880. The system of claim 859, wherein delegation occurs through an execution-instruction signal router.

881. The system of claim 880, wherein the execution-instruction signal router is a cross-point switch.

882. The system of claim 859, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

883. The system of claim 859, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

884. The system of claim 859, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

885. The system of claim 859, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

886. The system of claim 882, wherein a sleeping processing resource may be woken by a response to a request.

887. The system of claim 886, wherein the response is directed to a specific processing resource.

888. The system of claim 886, wherein the response indicates a once locked resource is unlocked.

889. The system of claim 859, wherein processing resources are communicatively disposed on a same die.

890. The system of claim 889, wherein an execution-instruction signal router is on the same die with processing resources.

891. The system of claim 859, wherein the binding names are stored by a linker.

892. The system of claim 859, further comprising,

means to replace the odd address value stored in an instruction stack with an instruction pointer at the even address, if late binding is being employed.

893. The system of claim 859, further comprising,

means to replace the odd address value stored in an instruction stack with an instruction pointer at the even address and replacing the odd address value in the binding name table with the even address, if dynamic binding is being employed.

894. A system of addressing registers, comprising:

means to determine if an execution-instruction operation-code provides additional register addressing information;

means to set cycle flags in a status register with the additional register addressing information;

means to examine an execution-instruction for a register address;

means to address a register specified by the register address and cycle flags.

895. The system of claim 894, wherein the system is completed within a single processing cycle.

896. The system of claim 894, wherein the execution-instruction is an execution-instruction signal for execution by a type of processing resource.

897. The system of claim 896, further comprising,

means to route the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

898. The system of claim 896, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

899. The system of claim 896, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

900. The system of claim 896, wherein the execution-instruction signal includes an operation-code.

901. The system of claim 900, wherein the operation-code indicates a type of resource on which to execute.

902. The system of claim 896, wherein the execution-instruction signal includes values from a status register.

903. The system of claim 896, wherein the execution-instruction signal includes values from a priority bit status register.

904. The system of claim 896, wherein the execution-instruction signal includes a processing resource identifier.

905. The system of claim 904, wherein the processing resource identifier is an integer processing unit number.

906. The system of claim 896, wherein the execution-instruction signal includes an address.

907. The system of claim 894, wherein an other processing resource may be an delegating processing resource.

908. The system of claim 894, wherein a processing resource is an integer processing unit.

909. The system of claim 894, wherein a processing resource is a mathematical processing unit.

910. The system of claim 894, wherein a processing resource is a memory management unit.

911. The system of claim 894, wherein a processing resource is a vector processing unit.

912. The system of claim 894, wherein a processing resource is a digital signal processing unit.

913. The system of claim 894, wherein a processing resource is a graphics processing unit.

914. The system of claim 894, wherein a processing resource is an input/output controller processing unit.

915. The system of claim 894, wherein a processing resource is an execution-instruction processing cache.

916. The system of claim 894, wherein delegation occurs through an execution-instruction signal router.

917. The system of claim 916, wherein the execution-instruction signal router is a cross-point switch.

918. The system of claim 894, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

919. The system of claim 894, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

920. The system of claim 894, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

921. The system of claim 894, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

922. The system of claim 918, wherein a sleeping processing resource may be woken by a response to a request.

923. The system of claim 922, wherein the response is directed to a specific processing resource.

924. The system of claim 922, wherein the response indicates a once locked resource is unlocked.

925. The system of claim 894, wherein processing resources are communicatively disposed on a same die.

926. The system of claim 925, wherein an execution-instruction signal router is on the same die with processing resources.

927. The system of claim 894, wherein specific operation-codes address extended register sets without requiring dedicated bits set aside for register addressing in the execution-instruction.

928. The system of claim 894, wherein the register address comprises 3 bits of the execution-instruction.

929. The system of claim 894, wherein a value of a cycle flag indicates that an operation-code is a multi-cycle operation.

930. A system of sharing memory, comprising:

means to obtain a request execution-instruction from a delegate processing resource at an instruction processing cache, wherein the request includes a target memory address;

means to determine if the target address is locked based on a lock variable at the target memory address;

means to determine what type of lock is provided in the request;

means to request that a requesting processing resource sleep until it is unlocked, if the target memory address is locked.

931. The system of claim 930, wherein the system is completed within a single processing cycle.

932. The system of claim 930, wherein the request execution-instruction is an execution-instruction signal for execution by a type of processing resource.

933. The system of claim 932, further comprising,

means to route the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

934. The system of claim 932, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

935. The system of claim 932, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

936. The system of claim 932, wherein the execution-instruction signal includes an operation-code.

937. The system of claim 936, wherein the operation-code indicates a type of resource on which to execute.

938. The system of claim 932, wherein the execution-instruction signal includes values from a status register.

939. The system of claim 932, wherein the execution-instruction signal includes values from a priority bit status register.

940. The system of claim 932, wherein the execution-instruction signal includes a processing resource identifier.

941. The system of claim 940, wherein the processing resource identifier is an integer processing unit number.

942. The system of claim 932, wherein the execution-instruction signal includes an address.

943. The system of claim 930, wherein an other processing resource may be an delegating processing resource.

944. The system of claim 930, wherein a processing resource is an integer processing unit.

945. The system of claim 930, wherein a processing resource is a mathematical processing unit.

946. The system of claim 930, wherein a processing resource is a memory management unit.

947. The system of claim 930, wherein a processing resource is a vector processing unit.

948. The system of claim 930, wherein a processing resource is a digital signal processing unit.

949. The system of claim 930, wherein a processing resource is a graphics processing unit.

950. The system of claim 930, wherein a processing resource is an input/output controller processing unit.

951. The system of claim 930, wherein a processing resource is an execution-instruction processing cache.

952. The system of claim 930, wherein delegation occurs through an execution-instruction signal router.

953. The system of claim 952, wherein the execution-instruction signal router is a cross-point switch.

954. The system of claim 930, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

955. The system of claim 930, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

956. The system of claim 930, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

957. The system of claim 930, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

958. The system of claim 954, wherein a sleeping processing resource may be woken by a response to a request.

959. The system of claim 958, wherein the response is directed to a specific processing resource.

960. The system of claim 958, wherein the response indicates a once locked resource is unlocked.

961. The system of claim 930, wherein processing resources are communicatively disposed on a same die.

962. The system of claim 961, wherein an execution-instruction signal router is on the same die with processing resources.

963. The system of claim 930, wherein the determination is made by the instruction processing cache's logic facilities.

964. The system of claim 930, wherein the requesting processing resource is unlocked when an unlock instruction is received that specifies the same target memory address.

965. The system of claim 930, wherein the target memory address is locked for read and write operations.

966. The system of claim 930, wherein the target memory address is locked for read only operations and the request specifies a read and write operation lock.

967. A system of sharing memory, comprising:

means to obtain a response to an execution-instruction from a delegate processing resource, wherein the response includes a target address at a processing resource;

means to determine if a target address is locked based on a lock variable at the target memory address;

means to determine value types by using a hash function;

means to update value types in a primary cache memory of a processing resource to a secondary cache memory for each value type, if each value type in a primary cache memory of a processing resource has not been updated to a secondary cache memory;

968. The system of claim 967, wherein the system is completed within a single processing cycle.

969. The system of claim 967, wherein the response is an execution-instruction signal for execution by a type of processing resource.

970. The system of claim 969, further comprising,

means to route the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

971. The system of claim 969, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

972. The system of claim 969, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

973. The system of claim 969, wherein the execution-instruction signal includes an operation-code.

974. The system of claim 973, wherein the operation-code indicates a type of resource on which to execute.

975. The system of claim 969, wherein the execution-instruction signal includes values from a status register.

976. The system of claim 969, wherein the execution-instruction signal includes values from a priority bit status register.

977. The system of claim 969, wherein the execution-instruction signal includes a processing resource identifier.

978. The system of claim 977, wherein the processing resource identifier is an integer processing unit number.

979. The system of claim 969, wherein the execution-instruction signal includes an address.

980. The system of claim 967, wherein an other processing resource may be an delegating processing resource.

981. The system of claim 967, wherein a processing resource is an integer processing unit.

982. The system of claim 967, wherein a processing resource is a mathematical processing unit.

983. The system of claim 967, wherein a processing resource is a memory management unit.

984. The system of claim 967, wherein a processing resource is a vector processing unit.

985. The system of claim 967, wherein a processing resource is a digital signal processing unit.

986. The system of claim 967, wherein a processing resource is a graphics processing unit.

987. The system of claim 967, wherein a processing resource is an input/output controller processing unit.

988. The system of claim 967, wherein a processing resource is an execution-instruction processing cache.

989. The system of claim 967, wherein delegation occurs through an execution-instruction signal router.

990. The system of claim 989, wherein the execution-instruction signal router is a cross-point switch.

991. The system of claim 967, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

992. The system of claim 967, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

993. The system of claim 967, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

994. The system of claim 967, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

995. The system of claim 991, wherein a sleeping processing resource may be woken by a response to a request.

996. The system of claim 995, wherein the response is directed to a specific processing resource.

997. The system of claim 995, wherein the response indicates a once locked resource is unlocked.

998. The system of claim 967, wherein processing resources are communicatively disposed on a same die.

999. The system of claim 998, wherein an execution-instruction signal router is on the same die with processing resources.

1000. The system of claim 967, wherein the response includes a target processing resource identifier.

1001. The system of claim 967, wherein the determination is made by a processing resource.

1002. The system of claim 967, wherein the value type is a global value.

1003. The system of claim 967, wherein the secondary cache memory is a Level 2 cache memory.

1004. The system of claim 967, wherein a primary cache memory is a Level 1 cache memory.

1005. The system of claim 967, further comprising,
means to marking a cache line of the updated value type as free.

1006. The system of claim 967, further comprising,
means to update lock variables in secondary cache memory for each updated value type.

1007. The system of claim 1006, wherein the lock variable updating is performed by the secondary cache memory, which is a processing cache memory.

1008. A system of sharing memory, comprising:
means to obtain a storage request including an event variable address and execution-instruction for a processing resource to sleep;
means to determine if an event variable is set;
means to provide a reply to a processing resource bus with the event variable address and event value, which will wake an appropriate processing resource from sleep, if the event value is set not to sleep;

means to set a sleep until event value for the event variable, if the event value is set to sleep.

1009. The system of claim 1008, wherein the system is completed within a single processing cycle.

1010. The system of claim 1008, wherein the storage request is an execution-instruction signal for execution by a type of processing resource.

1011. The system of claim 1010, further comprising,

means to route the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

1012. The system of claim 1010, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

1013. The system of claim 1010, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

1014. The system of claim 1010, wherein the execution-instruction signal includes an operation-code.

1015. The system of claim 1014, wherein the operation-code indicates a type of resource on which to execute.

1016. The system of claim 1010, wherein the execution-instruction signal includes values from a status register.

1017. The system of claim 1010, wherein the execution-instruction signal includes values from a priority bit status register.

1018. The system of claim 1010, wherein the execution-instruction signal includes a processing resource identifier.

1019. The system of claim 1018, wherein the processing resource identifier is an integer processing unit number.

1020. The system of claim 1010, wherein the execution-instruction signal includes an address.

1021. The system of claim 1008, wherein an other processing resource may be an delegating processing resource.

1022. The system of claim 1008, wherein a processing resource is an integer processing unit.

1023. The system of claim 1008, wherein a processing resource is a mathematical processing unit.

1024. The system of claim 1008, wherein a processing resource is a memory management unit.

1025. The system of claim 1008, wherein a processing resource is a vector processing unit.

1026. The system of claim 1008, wherein a processing resource is a digital signal processing unit.

1027. The system of claim 1008, wherein a processing resource is a graphics processing unit.

1028. The system of claim 1008, wherein a processing resource is an input/output controller processing unit.

1029. The system of claim 1008, wherein a processing resource is an execution-instruction processing cache.

1030. The system of claim 1008, wherein delegation occurs through an execution-instruction signal router.

1031. The system of claim 1030, wherein the execution-instruction signal router is a cross-point switch.

1032. The system of claim 1008, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

1033. The system of claim 1008, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

1034. The system of claim 1008, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

1035. The system of claim 1008, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

1036. The system of claim 1032, wherein a sleeping processing resource may be woken by a response to a request.

1037. The system of claim 1036, wherein the response is directed to a specific processing resource.

1038. The system of claim 1036, wherein the response indicates a once locked resource is unlocked.

1039. The system of claim 1008, wherein processing resources are communicatively disposed on a same die.

1040. The system of claim 1039, wherein an execution-instruction signal router is on the same die with processing resources.

1041. The system of claim 1008, wherein the determination is made at a processing cache.

1042. The system of claim 1008, wherein the event variable is set at the processing cache.

1043. The system of claim 1008, wherein the event variable is set in a register.

1044. The system of claim 1008, wherein the event variable is set in a target address in a processing cache.

1045. A system of sharing memory, comprising:

means to obtain a storage request including an event address at a processing cache;

means to combine a parameter with an event variable through in a processing cache;

means to provide a reply to a processing resource bus from the combination with the event address, which will wake an appropriate processing resource from sleep that is waiting on an event, if the event value is set not to sleep.

1046. The system of claim 1045, wherein the system is completed within a single processing cycle.

1047. The system of claim 1045, wherein the storage request is an execution-instruction signal for execution by a type of processing resource.

1048. The system of claim 1047, further comprising,

means to route the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

1049. The system of claim 1047, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

1050. The system of claim 1047, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

1051. The system of claim 1047, wherein the execution-instruction signal includes an operation-code.

1052. The system of claim 1051, wherein the operation-code indicates a type of resource on which to execute.

1053. The system of claim 1047, wherein the execution-instruction signal includes values from a status register.

1054. The system of claim 1047, wherein the execution-instruction signal includes values from a priority bit status register.

1055. The system of claim 1047, wherein the execution-instruction signal includes a processing resource identifier.

1056. The system of claim 1055, wherein the processing resource identifier is an integer processing unit number.

1057. The system of claim 1047, wherein the execution-instruction signal includes an address.

1058. The system of claim 1045, wherein an other processing resource may be an delegating processing resource.

1059. The system of claim 1045, wherein a processing resource is an integer processing unit.

1060. The system of claim 1045, wherein a processing resource is a mathematical processing unit.

1061. The system of claim 1045, wherein a processing resource is a memory management unit.

1062. The system of claim 1045, wherein a processing resource is a vector processing unit.

1063. The system of claim 1045, wherein a processing resource is a digital signal processing unit.

1064. The system of claim 1045, wherein a processing resource is a graphics processing unit.

1065. The system of claim 1045, wherein a processing resource is an input/output controller processing unit.

1066. The system of claim 1045, wherein a processing resource is an execution-instruction processing cache.

1067. The system of claim 1045, wherein delegation occurs through an execution-instruction signal router.

1068. The system of claim 1067, wherein the execution-instruction signal router is a cross-point switch.

1069. The system of claim 1045, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

1070. The system of claim 1045, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

1071. The system of claim 1045, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

1072. The system of claim 1045, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

1073. The system of claim 1069, wherein a sleeping processing resource may be woken by a response to a request.

1074. The system of claim 1073, wherein the response is directed to a specific processing resource.

1075. The system of claim 1073, wherein the response indicates a once locked resource is unlocked.

1076. The system of claim 1045, wherein processing resources are communicatively disposed on a same die.

1077. The system of claim 1076, wherein an execution-instruction signal router is on the same die with processing resources.

1078. The system of claim 1045, wherein the parameter is a bit-mask.

1079. The system of claim 1045, wherein the combination is a logical-OR.

1080. A system of sharing memory, comprising:

means to obtain a storage request including a semaphore address at a processing cache;

means to set a semaphore variable in a processing cache at the semaphore address;

means to provide an operating system trap-call to wait on a processing queue, if the storage request includes a wait-on-semaphore instruction;

means to provide an operating system trap-call to signal on a processing queue, if the storage request includes a signal-on-semaphore instruction;

1081. The system of claim 1080, wherein the system is completed within a single processing cycle.

1082. The system of claim 1080, wherein the storage request is an execution-instruction signal for execution by a type of processing resource.

1083. The system of claim 1082, further comprising,

means to route the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

1084. The system of claim 1082, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

1085. The system of claim 1082, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

1086. The system of claim 1082, wherein the execution-instruction signal includes an operation-code.

1087. The system of claim 1086, wherein the operation-code indicates a type of resource on which to execute.

1088. The system of claim 1082, wherein the execution-instruction signal includes values from a status register.

1089. The system of claim 1082, wherein the execution-instruction signal includes values from a priority bit status register.

1090. The system of claim 1082, wherein the execution-instruction signal includes a processing resource identifier.

1091. The system of claim 1090, wherein the processing resource identifier is an integer processing unit number.

1092. The system of claim 1082, wherein the execution-instruction signal includes an address.

1093. The system of claim 1080, wherein an other processing resource may be an delegating processing resource.

1094. The system of claim 1080, wherein a processing resource is an integer processing unit.

1095. The system of claim 1080, wherein a processing resource is a mathematical processing unit.

1096. The system of claim 1080, wherein a processing resource is a memory management unit.

1097. The system of claim 1080, wherein a processing resource is a vector processing unit.

1098. The system of claim 1080, wherein a processing resource is a digital signal processing unit.

1099. The system of claim 1080, wherein a processing resource is a graphics processing unit.

1100. The system of claim 1080, wherein a processing resource is an input/output controller processing unit.

1101. The system of claim 1080, wherein a processing resource is an execution-instruction processing cache.

1102. The system of claim 1080, wherein delegation occurs through an execution-instruction signal router.

1103. The system of claim 1102, wherein the execution-instruction signal router is a cross-point switch.

1104. The system of claim 1080, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

1105. The system of claim 1080, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

1106. The system of claim 1080, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

1107. The system of claim 1080, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

1108. The system of claim 1104, wherein a sleeping processing resource may be woken by a response to a request.

1109. The system of claim 1108, wherein the response is directed to a specific processing resource.

1110. The system of claim 1108, wherein the response indicates a once locked resource is unlocked.

1111. The system of claim 1080, wherein processing resources are communicatively disposed on a same die.

1112. The system of claim 1111, wherein an execution-instruction signal router is on the same die with processing resources.

1113. The system of claim 1080, wherein the trap-call causes rescheduling so that other processes can run on a particular processing resource.

1114. The system of claim 1080, wherein other processes include threads.

1115. The system of claim 1080, wherein other processes include threads.

1116. An apparatus of register addressing, comprising:
a memory, the memory for storing instructions;

a processing resource that may issue a plurality of processing instructions stored in the memory, wherein the instructions issue signals to:

obtain a register bank address;

decode an operation-code;

set a cycle flag in a status register based on the decoding of the operation-code;

access a register addressed by the cycle flag within the addressed register bank.

1117. The apparatus of claim 1116, further comprising,

employ a set number of bits in an execution-instruction to establish the number of registers within a register bank.

1118. The apparatus of claim 1116, further comprising,

employ a set number of bits in an execution-instruction to establish the number of register banks.

1119. An apparatus of setting intra-processor processing resources to sleep, comprising:

set processing resources that issue requests to delegating processing resources to sleep until a response is provided;

set processing resources waiting on shared and locked memory being accessed by other processing resources to sleep until the memory is unlocked.

1120. An apparatus of execution-instruction delegation between processing resources, comprising:

a memory, the memory for storing instructions;

a processing resource that may issue a plurality of processing instructions stored in the memory, wherein the instructions issue signals to:

share memory between processing resources,

wherein the memory itself includes instruction execution logic,

wherein the processing resources are communicatively accessible through an instruction execution router;

wherein the processing resources are on the same die;

delegate execution-instructions from originating processing resources to other processing resources.

1121. The apparatus of claim 1120, wherein the system is completed within a single processing cycle.

1122. The apparatus of claim 1120, wherein the other processing resource may be the originating processing resource.

1123. The apparatus of claim 1120, wherein a processing resource is an integer processing unit.

1124. The apparatus of claim 1120, wherein a processing resource is a mathematical processing unit.

1125. The apparatus of claim 1120, wherein a processing resource is a memory management unit.

1126. The apparatus of claim 1120, wherein a processing resource is a vector processing unit.

1127. The apparatus of claim 1120, wherein a processing resource is a digital signal processing unit.

1128. The apparatus of claim 1120, wherein a processing resource is a graphics processing unit.

1129. The apparatus of claim 1120, wherein a processing resource is an input/output controller processing unit.

1130. The apparatus of claim 1120, wherein a processing resource is an execution-instruction processing cache.

1131. The apparatus of claim 1120, wherein delegation occurs through an execution-instruction signal router.

1132. The apparatus of claim 1131, wherein the execution-instruction signal router is a cross-point switch.

1133. The apparatus of claim 1120, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

1134. The apparatus of claim 1120, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

1135. The apparatus of claim 1120, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

1136. The apparatus of claim 1120, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

1137. The apparatus of claim 1133, wherein a sleeping processing resource may be woken by a response to a request.

1138. The apparatus of claim 1137, wherein the response is directed to a specific processing resource.

1139. The apparatus of claim 1137, wherein the response indicates a once locked resource is unlocked.

1140. The apparatus of claim 1120, wherein processing resources are communicatively disposed on a same die.

1141. The apparatus of claim 1140, wherein an execution-instruction signal router is on the same die with processing resources.

1142. An apparatus of execution-instruction delegation between processing resources, comprising:

a memory, the memory for storing instructions;

a processing resource that may issue a plurality of processing instructions stored in the memory, wherein the instructions issue signals to:

obtain an execution instruction, wherein the execution instruction is obtained at a processing resource;

determine whether an operation-code within the execution instruction should be delegated to an other processing resource;

execute the execution instruction, if the operation-code within the execution instruction should not be delegated to an other processing resource;

route the execution instruction to an other processing resource, if the operation-code within the execution instruction is for an other processing resource.

1143. The apparatus of claim 1142, wherein the system is completed within a single processing cycle.

1144. The apparatus of claim 1142, wherein the routed execution instruction is a request for execution by a type of processing resource.

1145. The apparatus of claim 1144, wherein the request includes an operation-code.

1146. The apparatus of claim 1145, wherein the operation-code indicates a type of resource on which to execute.

1147. The apparatus of claim 1144, wherein the request includes values from a status register.

1148. The apparatus of claim 1144, wherein the request includes values from a priority bit status register.

1149. The apparatus of claim 1144, wherein the request includes a processing resource identifier.

1150. The apparatus of claim 1149, wherein the processing resource identifier is an integer processing unit number.

1151. The apparatus of claim 1144, wherein the request is obtained by an execution-instruction signal router.

1152. The apparatus of claim 1144, wherein the request includes an address.

1153. The apparatus of claim 1142, wherein the operation-code indicates a type of resource on which to execute.

1154. The apparatus of claim 1142, wherein the other processing resource may be the originating processing resource.

1155. The apparatus of claim 1142, wherein a processing resource is an integer processing unit.

1156. The apparatus of claim 1142, wherein a processing resource is a mathematical processing unit.

1157. The apparatus of claim 1142, wherein a processing resource is a memory management unit.

1158. The apparatus of claim 1142, wherein a processing resource is a vector processing unit.

1159. The apparatus of claim 1142, wherein a processing resource is a digital signal processing unit.

1160. The apparatus of claim 1142, wherein a processing resource is a graphics processing unit.

1161. The apparatus of claim 1142, wherein a processing resource is an input/output controller processing unit.

1162. The apparatus of claim 1142, wherein a processing resource is an execution-instruction processing cache.

1163. The apparatus of claim 1142, wherein routing occurs through an execution-instruction signal router.

1164. The apparatus of claim 1163, wherein the execution-instruction signal router is a cross-point switch.

1165. The apparatus of claim 1142, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

1166. The apparatus of claim 1142, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

1167. The apparatus of claim 1142, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

1168. The apparatus of claim 1142, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

1169. The apparatus of claim 1142, wherein processing resources are communicatively disposed on a same die.

1170. The apparatus of claim 1169, wherein an execution-instruction signal router is on the same die with processing resources.

1171. An apparatus of execution-instruction delegation between processing resources, comprising:

a memory, the memory for storing instructions;

a processing resource that may issue a plurality of processing instructions stored in the memory, wherein the instructions issue signals to:

obtain an request signal, wherein the request signal is from a requesting processing resource;

determine a priority dead-lock-avoidance value, wherein the priority dead-lock value is used to select among multiple requests with equal priority;

examine a request priority value for each submitted request;

select a request with a highest priority value, if more than one processing resource requests a same target resource;

provide a selected request to a target processing resource.

1172. The apparatus of claim 1171, wherein the system is completed within a single processing cycle.

1173. The apparatus of claim 1171, wherein the request signal is a request for execution by a type of processing resource.

1174. The apparatus of claim 1173, further comprising,

route the request to an other processing resource, if an operation-code within the request is for an other processing resource.

1175. The apparatus of claim 1173, wherein the request is obtained at an execution-instruction signal router.

1176. The apparatus of claim 1173, wherein the execution-instruction signal is provided by a delegating processing resource via an execution-instruction signal router.

1177. The apparatus of claim 1173, wherein the request includes an operation-code.

1178. The apparatus of claim 1177, wherein the operation-code indicates a type of resource on which to execute.

1179. The apparatus of claim 1173, wherein the request includes values from a status register.

1180. The apparatus of claim 1173, wherein the request includes values from a priority bit status register.

1181. The apparatus of claim 1173, wherein the request includes a processing resource identifier.

1182. The apparatus of claim 1181, wherein the processing resource identifier is an integer processing unit number.

1183. The apparatus of claim 1173, wherein the request includes an address.

1184. The apparatus of claim 1171, wherein the other processing resource may be the originating processing resource.

1185. The apparatus of claim 1171, wherein a processing resource is an integer processing unit.

1186. The apparatus of claim 1171, wherein a processing resource is a mathematical processing unit.

1187. The apparatus of claim 1171, wherein a processing resource is a memory management unit.

1188. The apparatus of claim 1171, wherein a processing resource is a vector processing unit.

1189. The apparatus of claim 1171, wherein a processing resource is a digital signal processing unit.

1190. The apparatus of claim 1171, wherein a processing resource is a graphics processing unit.

1191. The apparatus of claim 1171, wherein a processing resource is an input/output controller processing unit.

1192. The apparatus of claim 1171, wherein a processing resource is an execution-instruction processing cache.

1193. The apparatus of claim 1171, wherein delegation occurs through an execution-instruction signal router.

1194. The apparatus of claim 1193, wherein the execution-instruction signal router is a cross-point switch.

1195. The apparatus of claim 1171, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

1196. The apparatus of claim 1171, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

1197. The apparatus of claim 1171, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

1198. The apparatus of claim 1171, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

1199. The apparatus of claim 1195, wherein a sleeping processing resource may be woken by a response to a request.

1200. The apparatus of claim 1199, wherein the response is directed to a specific processing resource.

1201. The apparatus of claim 1199, wherein the response indicates a once locked resource is unlocked.

1202. The apparatus of claim 1171, wherein processing resources are communicatively disposed on a same die.

1203. The apparatus of claim 1202, wherein an execution-instruction signal router is on the same die with processing resources.

1204. The apparatus of claim 1171, wherein the priority dead-lock-avoidance value is iterated.

1205. The apparatus of claim 1171, further comprising,

select a request, if not more than one processing resource requests a same target resource.

1206. The apparatus of claim 1171, further comprising,

select the request with a highest priority value, if not more than one request has a same highest priority value;

select a request based on the priority dead-lock-avoidance value, if more than one request has a same highest priority value.

1207. The apparatus of claim 1171, further comprising,

provide a request granted acknowledgement to the requesting processing resource whose request was selected.

1208. The apparatus of claim 1171, further comprising,

clear the operation-code in the requesting processing resource whose request was selected.

1209. An apparatus of execution-instruction delegation between processing resources, comprising:

a memory, the memory for storing instructions;

a processing resource that may issue a plurality of processing instructions stored in the memory, wherein the instructions issue signals to:

process a request execution-instruction, wherein the request execution-instruction includes a requesting processing resource identifier;

prepare a response into a result register, wherein the response includes the requesting processing resource identifier;

present the response to all processing resources.

1210. The apparatus of claim 1209, wherein the system is completed within a single processing cycle.

1211. The apparatus of claim 1209, wherein the request execution-instruction is a request for execution by a type of processing resource.

1212. The apparatus of claim 1211, further comprising,

route the request to an other processing resource, if an operation-code within the request is for an other processing resource.

1213. The apparatus of claim 1211, wherein the request is obtained at an execution-instruction signal router.

1214. The apparatus of claim 1211, wherein the execution-instruction signal is provided by a delegating processing resource via an execution-instruction signal router.

1215. The apparatus of claim 1211, wherein the request includes an operation-code.

1216. The apparatus of claim 1215, wherein the operation-code indicates a type of resource on which to execute.

1217. The apparatus of claim 1211, wherein the request includes values from a status register.

1218. The apparatus of claim 1211, wherein the request includes values from a priority bit status register.

1219. The apparatus of claim 1211, wherein the request includes a processing resource identifier.

1220. The apparatus of claim 1219, wherein the processing resource identifier is an integer processing unit number.

1221. The apparatus of claim 1211, wherein the request includes an address.

1222. The apparatus of claim 1209, wherein the other processing resource may be the originating processing resource.

1223. The apparatus of claim 1209, wherein a processing resource is an integer processing unit.

1224. The apparatus of claim 1209, wherein a processing resource is a mathematical processing unit.

1225. The apparatus of claim 1209, wherein a processing resource is a memory management unit.

1226. The apparatus of claim 1209, wherein a processing resource is a vector processing unit.

1227. The apparatus of claim 1209, wherein a processing resource is a digital signal processing unit.

1228. The apparatus of claim 1209, wherein a processing resource is a graphics processing unit.

1229. The apparatus of claim 1209, wherein a processing resource is an input/output controller processing unit.

1230. The apparatus of claim 1209, wherein a processing resource is an execution-instruction processing cache.

1231. The apparatus of claim 1209, wherein delegation occurs through an execution-instruction signal router.

1232. The apparatus of claim 1231, wherein the execution-instruction signal router is a cross-point switch.

1233. The apparatus of claim 1209, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

1234. The apparatus of claim 1209, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

1235. The apparatus of claim 1209, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

1236. The apparatus of claim 1209, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

1237. The apparatus of claim 1233, wherein a sleeping processing resource may be woken by a response to a request.

1238. The apparatus of claim 1237, wherein the response is directed to a specific processing resource.

1239. The apparatus of claim 1237, wherein the response indicates a once locked resource is unlocked.

1240. The apparatus of claim 1209, wherein processing resources are communicatively disposed on a same die.

1241. The apparatus of claim 1240, wherein an execution-instruction signal router is on the same die with processing resources.

1242. An apparatus of execution-instruction delegation between processing resources, comprising:

a memory, the memory for storing instructions;

a processing resource that may issue a plurality of processing instructions stored in the memory, wherein the instructions issue signals to:

obtain a result execution-instruction from a delegate processing resource, wherein the result includes a requesting processing resource identifier;

wake the requesting processing resource, if the requesting processing resource identifier identifies the instant processing resource;

wake a processing resource, if a processing resource is waiting to be unlocked and if the obtained result includes an unlock flag and if an address in the result matches a locked address of an instant processing resource.

1243. The apparatus of claim 1242, wherein the system is completed within a single processing cycle.

1244. The apparatus of claim 1242, wherein the result execution-instruction is an execution-instruction signal for execution by a type of processing resource.

1245. The apparatus of claim 1244, further comprising,

route the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

1246. The apparatus of claim 1244, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

1247. The apparatus of claim 1244, wherein the execution-instruction signal is provided by a delegating processing resource via an execution-instruction signal router.

1248. The apparatus of claim 1244, wherein the execution-instruction signal includes an operation-code.

1249. The apparatus of claim 1248, wherein the operation-code indicates a type of resource on which to execute.

1250. The apparatus of claim 1244, wherein the execution-instruction signal includes values from a status register.

1251. The apparatus of claim 1244, wherein the execution-instruction signal includes values from a priority bit status register.

1252. The apparatus of claim 1244, wherein the execution-instruction signal includes a processing resource identifier.

1253. The apparatus of claim 1252, wherein the processing resource identifier is an integer processing unit number.

1254. The apparatus of claim 1244, wherein the execution-instruction signal includes an address.

1255. The apparatus of claim 1242, wherein an other processing resource may be an delegating processing resource.

1256. The apparatus of claim 1242, wherein a processing resource is an integer processing unit.

1257. The apparatus of claim 1242, wherein a processing resource is a mathematical processing unit.

1258. The apparatus of claim 1242, wherein a processing resource is a memory management unit.

1259. The apparatus of claim 1242, wherein a processing resource is a vector processing unit.

1260. The apparatus of claim 1242, wherein a processing resource is a digital signal processing unit.

1261. The apparatus of claim 1242, wherein a processing resource is a graphics processing unit.

1262. The apparatus of claim 1242, wherein a processing resource is an input/output controller processing unit.

1263. The apparatus of claim 1242, wherein a processing resource is an execution-instruction processing cache.

1264. The apparatus of claim 1242, wherein delegation occurs through an execution-instruction signal router.

1265. The apparatus of claim 1264, wherein the execution-instruction signal router is a cross-point switch.

1266. The apparatus of claim 1242, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

1267. The apparatus of claim 1242, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

1268. The apparatus of claim 1242, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

1269. The apparatus of claim 1242, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

1270. The apparatus of claim 1266, wherein a sleeping processing resource may be woken by a response to a request.

1271. The apparatus of claim 1270, wherein the response is directed to a specific processing resource.

1272. The apparatus of claim 1270, wherein the response indicates a once locked resource is unlocked.

1273. The apparatus of claim 1242, wherein processing resources are communicatively disposed on a same die.

1274. The apparatus of claim 1273, wherein an execution-instruction signal router is on the same die with processing resources.

1275. The apparatus of claim 1242, further comprising,

execute a next execution-instruction, if the requesting processing resource identifier identifies the instant processing resource;

execute an execution-instruction that failed to execute, if a processing resource is waiting to be unlocked and if the obtained result includes an unlock flag and if an address in the result matches a locked address of an instant processing resource.

1276. An apparatus of memory access optimization, comprising:

a memory, the memory for storing instructions;

a processing resource that may issue a plurality of processing instructions stored in the memory, wherein the instructions issue signals to:

obtain a storage-access request from a delegated processing resource, wherein the request includes a target memory address;

determine the target memory address from the request;

compare the target memory address with register values, wherein register values are used to establish a data type of the target memory address for subsequent storage in an apportioned region of cache memory;

obtain information from the target memory address;

store the information in an apportioned region of cache memory.

1277. The apparatus of claim 1276, wherein the system is completed within a single processing cycle.

1278. The apparatus of claim 1276, wherein the storage-access request is an execution-instruction signal for execution by a type of processing resource.

1279. The apparatus of claim 1278, further comprising,

route the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

1280. The apparatus of claim 1278, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

1281. The apparatus of claim 1278, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

1282. The apparatus of claim 1278, wherein the execution-instruction signal includes an operation-code.

1283. The apparatus of claim 1282, wherein the operation-code indicates a type of resource on which to execute.

1284. The apparatus of claim 1278, wherein the execution-instruction signal includes values from a status register.

1285. The apparatus of claim 1278, wherein the execution-instruction signal includes values from a priority bit status register.

1286. The apparatus of claim 1278, wherein the execution-instruction signal includes a processing resource identifier.

1287. The apparatus of claim 1286, wherein the processing resource identifier is an integer processing unit number.

1288. The apparatus of claim 1278, wherein the execution-instruction signal includes an address.

1289. The apparatus of claim 1276, wherein an other processing resource may be an delegating processing resource.

1290. The apparatus of claim 1276, wherein a processing resource is an integer processing unit.

1291. The apparatus of claim 1276, wherein a processing resource is a mathematical processing unit.

1292. The apparatus of claim 1276, wherein a processing resource is a memory management unit.

1293. The apparatus of claim 1276, wherein a processing resource is a vector processing unit.

1294. The apparatus of claim 1276, wherein a processing resource is a digital signal processing unit.

1295. The apparatus of claim 1276, wherein a processing resource is a graphics processing unit.

1296. The apparatus of claim 1276, wherein a processing resource is an input/output controller processing unit.

1297. The apparatus of claim 1276, wherein a processing resource is an execution-instruction processing cache.

1298. The apparatus of claim 1276, wherein delegation occurs through an execution-instruction signal router.

1299. The apparatus of claim 1298, wherein the execution-instruction signal router is a cross-point switch.

1300. The apparatus of claim 1276, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

1301. The apparatus of claim 1276, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

1302. The apparatus of claim 1276, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

1303. The apparatus of claim 1276, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

1304. The apparatus of claim 1300, wherein a sleeping processing resource may be woken by a response to a request.

1305. The apparatus of claim 1304, wherein the response is directed to a specific processing resource.

1306. The apparatus of claim 1304, wherein the response indicates a once locked resource is unlocked.

1307. The apparatus of claim 1276, wherein processing resources are communicatively disposed on a same die.

1308. The apparatus of claim 1307, wherein an execution-instruction signal router is on the same die with processing resources.

1309. The apparatus of claim 1276, wherein the storage-access request is obtained at a processing resource.

1310. The apparatus of claim 1276, wherein information at the target memory address is designated to be local data if the target memory address is within an address range established by stack base and end register values.

1311. The apparatus of claim 1276, wherein information at the target memory address is designated to be global data if the target memory address is outside an address range established by stack base and end register values.

1312. The apparatus of claim 1276, wherein an apportionment is designated for each type of data for optimized access.

1313. The apparatus of claim 1276, wherein the apportioned region is a hash region.

1314. The apparatus of claim 1313, wherein a hash region is designated for local data.

1315. The apparatus of claim 1313, wherein a hash region is designated for global data.

1316. The apparatus of claim 1313, wherein the apportionment for a local data region is greater than a region for global data.

1317. An apparatus of reduced size execution of execution-instructions, comprising:

a memory, the memory for storing instructions;

a processing resource that may issue a plurality of processing instructions stored in the memory, wherein the instructions issue signals to:

obtain a an execution-instruction at a processing resource;

append a literal constant in a literal register, if a literal flag is set by examining a special register and if there is literal prefix in the execution-instruction;

execute an execution-instruction with a constant in a literal register, if a literal flag is set by examining a special register and if there is no literal prefix in the execution-instruction;

set a literal constant flag in a status register and placing a literal constant in a literal register, if a literal flag is not set by examining a special register and if there is literal prefix in the execution-instruction;

execute an execution-instruction, if a literal flag is not set by examining a special register and if there is no literal prefix in the execution-instruction.

1318. The apparatus of claim 1317, wherein the system is completed within a single processing cycle.

1319. The apparatus of claim 1317, wherein the execution-instruction is an execution-instruction signal for execution by a type of processing resource.

1320. The apparatus of claim 1319, further comprising,

route the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

1321. The apparatus of claim 1319, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

1322. The apparatus of claim 1319, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

1323. The apparatus of claim 1319, wherein the execution-instruction signal includes an operation-code.

1324. The apparatus of claim 1323, wherein the operation-code indicates a type of resource on which to execute.

1325. The apparatus of claim 1319, wherein the execution-instruction signal includes values from a status register.

1326. The apparatus of claim 1319, wherein the execution-instruction signal includes values from a priority bit status register.

1327. The apparatus of claim 1319, wherein the execution-instruction signal includes a processing resource identifier.

1328. The apparatus of claim 1327, wherein the processing resource identifier is an integer processing unit number.

1329. The apparatus of claim 1319, wherein the execution-instruction signal includes an address.

1330. The apparatus of claim 1317, wherein an other processing resource may be an delegating processing resource.

1331. The apparatus of claim 1317, wherein a processing resource is an integer processing unit.

1332. The apparatus of claim 1317, wherein a processing resource is a mathematical processing unit.

1333. The apparatus of claim 1317, wherein a processing resource is a memory management unit.

1334. The apparatus of claim 1317, wherein a processing resource is a vector processing unit.

1335. The apparatus of claim 1317, wherein a processing resource is a digital signal processing unit.

1336. The apparatus of claim 1317, wherein a processing resource is a graphics processing unit.

1337. The apparatus of claim 1317, wherein a processing resource is an input/output controller processing unit.

1338. The apparatus of claim 1317, wherein a processing resource is an execution-instruction processing cache.

1339. The apparatus of claim 1317, wherein delegation occurs through an execution-instruction signal router.

1340. The apparatus of claim 1339, wherein the execution-instruction signal router is a cross-point switch.

1341. The apparatus of claim 1317, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

1342. The apparatus of claim 1317, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

1343. The apparatus of claim 1317, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

1344. The apparatus of claim 1317, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

1345. The apparatus of claim 1341, wherein a sleeping processing resource may be woken by a response to a request.

1346. The apparatus of claim 1345, wherein the response is directed to a specific processing resource.

1347. The apparatus of claim 1345, wherein the response indicates a once locked resource is unlocked.

1348. The apparatus of claim 1317, wherein processing resources are communicatively disposed on a same die.

1349. The apparatus of claim 1348, wherein an execution-instruction signal router is on the same die with processing resources.

1350. The apparatus of claim 1317, wherein the literal constant is appended by employing a shift.

1351. The apparatus of claim 1317, wherein the processing resource appends the literal constant.

1352. The apparatus of claim 1317, wherein the execution-instruction is executed as an extended execution of an operation-code.

1353. The apparatus of claim 1352, wherein a processing resource executes the execution-instruction.

1354. The apparatus of claim 1317, wherein the status register is in the processing resource.

1355. The apparatus of claim 1317, wherein the literal register is in the processing resource.

1356. The apparatus of claim 1317, wherein the execution-instruction is executed as a non-extended execution of an operation-code.

1357. The apparatus of claim 1356, wherein a processing resource executes the execution-instruction.

1358. An apparatus of binding instructions, comprising:

a memory, the memory for storing instructions;

a processing resource that may issue a plurality of processing instructions stored in the memory, wherein the instructions issue signals to:

store binding names at odd addresses;

generate a binding name interrupt, if a processing resource attempts to load an instruction register with an odd address value;

provide an operating system with a binding name interrupt;

perform a look-up of the odd address value that generated the binding name interrupt in a binding name table, which was established by linker;

replace the odd address value in an instruction register with an even address found in the binding name table.

1359. The apparatus of claim 1358, wherein the interrupt is an execution-instruction signal for execution by a type of processing resource.

1360. The apparatus of claim 1359, further comprising,

route the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

1361. The apparatus of claim 1359, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

1362. The apparatus of claim 1359, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

1363. The apparatus of claim 1359, wherein the execution-instruction signal includes an operation-code.

1364. The apparatus of claim 1363, wherein the operation-code indicates a type of resource on which to execute.

1365. The apparatus of claim 1359, wherein the execution-instruction signal includes values from a status register.

1366. The apparatus of claim 1359, wherein the execution-instruction signal includes values from a priority bit status register.

1367. The apparatus of claim 1359, wherein the execution-instruction signal includes a processing resource identifier.

1368. The apparatus of claim 1367, wherein the processing resource identifier is an integer processing unit number.

1369. The apparatus of claim 1359, wherein the execution-instruction signal includes an address.

1370. The apparatus of claim 1358, wherein an other processing resource may be an delegating processing resource.

1371. The apparatus of claim 1358, wherein a processing resource is an integer processing unit.

1372. The apparatus of claim 1358, wherein a processing resource is a mathematical processing unit.

1373. The apparatus of claim 1358, wherein a processing resource is a memory management unit.

1374. The apparatus of claim 1358, wherein a processing resource is a vector processing unit.

1375. The apparatus of claim 1358, wherein a processing resource is a digital signal processing unit.

1376. The apparatus of claim 1358, wherein a processing resource is a graphics processing unit.

1377. The apparatus of claim 1358, wherein a processing resource is an input/output controller processing unit.

1378. The apparatus of claim 1358, wherein a processing resource is an execution-instruction processing cache.

1379. The apparatus of claim 1358, wherein delegation occurs through an execution-instruction signal router.

1380. The apparatus of claim 1379, wherein the execution-instruction signal router is a cross-point switch.

1381. The apparatus of claim 1358, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

1382. The apparatus of claim 1358, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

1383. The apparatus of claim 1358, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

1384. The apparatus of claim 1358, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

1385. The apparatus of claim 1381, wherein a sleeping processing resource may be woken by a response to a request.

1386. The apparatus of claim 1385, wherein the response is directed to a specific processing resource.

1387. The apparatus of claim 1385, wherein the response indicates a once locked resource is unlocked.

1388. The apparatus of claim 1358, wherein processing resources are communicatively disposed on a same die.

1389. The apparatus of claim 1388, wherein an execution-instruction signal router is on the same die with processing resources.

1390. The apparatus of claim 1358, wherein the binding names are stored by a linker.

1391. The apparatus of claim 1358, further comprising,

replace the odd address value stored in an instruction stack with an instruction pointer at the even address, if late binding is being employed.

1392. The apparatus of claim 1358, further comprising,

replace the odd address value stored in an instruction stack with an instruction pointer at the even address and replacing the odd address value in the binding name table with the even address, if dynamic binding is being employed.

1393. An apparatus of addressing registers, comprising:

a memory, the memory for storing instructions;

a processing resource that may issue a plurality of processing instructions stored in the memory, wherein the instructions issue signals to:

determine if an execution-instruction operation-code provides additional register addressing information;

set cycle flags in a status register with the additional register addressing information;

examine an execution-instruction for a register address;

address a register specified by the register address and cycle flags.

1394. The apparatus of claim 1393, wherein the system is completed within a single processing cycle.

1395. The apparatus of claim 1393, wherein the execution-instruction is an execution-instruction signal for execution by a type of processing resource.

1396. The apparatus of claim 1395, further comprising,

route the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

1397. The apparatus of claim 1395, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

1398. The apparatus of claim 1395, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

1399. The apparatus of claim 1395, wherein the execution-instruction signal includes an operation-code.

1400. The apparatus of claim 1399, wherein the operation-code indicates a type of resource on which to execute.

1401. The apparatus of claim 1395, wherein the execution-instruction signal includes values from a status register.

1402. The apparatus of claim 1395, wherein the execution-instruction signal includes values from a priority bit status register.

1403. The apparatus of claim 1395, wherein the execution-instruction signal includes a processing resource identifier.

1404. The apparatus of claim 1403, wherein the processing resource identifier is an integer processing unit number.

1405. The apparatus of claim 1395, wherein the execution-instruction signal includes an address.

1406. The apparatus of claim 1393, wherein an other processing resource may be an delegating processing resource.

1407. The apparatus of claim 1393, wherein a processing resource is an integer processing unit.

1408. The apparatus of claim 1393, wherein a processing resource is a mathematical processing unit.

1409. The apparatus of claim 1393, wherein a processing resource is a memory management unit.

1410. The apparatus of claim 1393, wherein a processing resource is a vector processing unit.

1411. The apparatus of claim 1393, wherein a processing resource is a digital signal processing unit.

1412. The apparatus of claim 1393, wherein a processing resource is a graphics processing unit.

1413. The apparatus of claim 1393, wherein a processing resource is an input/output controller processing unit.

1414. The apparatus of claim 1393, wherein a processing resource is an execution-instruction processing cache.

1415. The apparatus of claim 1393, wherein delegation occurs through an execution-instruction signal router.

1416. The apparatus of claim 1415, wherein the execution-instruction signal router is a cross-point switch.

1417. The apparatus of claim 1393, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

1418. The apparatus of claim 1393, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

1419. The apparatus of claim 1393, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

1420. The apparatus of claim 1393, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

1421. The apparatus of claim 1417, wherein a sleeping processing resource may be woken by a response to a request.

1422. The apparatus of claim 1421, wherein the response is directed to a specific processing resource.

1423. The apparatus of claim 1421, wherein the response indicates a once locked resource is unlocked.

1424. The apparatus of claim 1393, wherein processing resources are communicatively disposed on a same die.

1425. The apparatus of claim 1424, wherein an execution-instruction signal router is on the same die with processing resources.

1426. The apparatus of claim 1393, wherein specific operation-codes address extended register sets without requiring dedicated bits set aside for register addressing in the execution-instruction.

1427. The apparatus of claim 1393, wherein the register address comprises 3 bits of the execution-instruction.

1428. The apparatus of claim 1393, wherein a value of a cycle flag indicates that an operation-code is a multi-cycle operation.

1429. An apparatus of sharing memory, comprising:

a memory, the memory for storing instructions;

a processing resource that may issue a plurality of processing instructions stored in the memory, wherein the instructions issue signals to:

obtain a request execution-instruction from a delegate processing resource at an instruction processing cache, wherein the request includes a target memory address;

determine if the target address is locked based on a lock variable at the target memory address;

determine what type of lock is provided in the request;

request that a requesting processing resource sleep until it is unlocked, if the target memory address is locked.

1430. The apparatus of claim 1429, wherein the system is completed within a single processing cycle.

1431. The apparatus of claim 1429, wherein the request execution-instruction is an execution-instruction signal for execution by a type of processing resource.

1432. The apparatus of claim 1431, further comprising,

route the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

1433. The apparatus of claim 1431, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

1434. The apparatus of claim 1431, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

1435. The apparatus of claim 1431, wherein the execution-instruction signal includes an operation-code.

1436. The apparatus of claim 1435, wherein the operation-code indicates a type of resource on which to execute.

1437. The apparatus of claim 1431, wherein the execution-instruction signal includes values from a status register.

1438. The apparatus of claim 1431, wherein the execution-instruction signal includes values from a priority bit status register.

1439. The apparatus of claim 1431, wherein the execution-instruction signal includes a processing resource identifier.

1440. The apparatus of claim 1439, wherein the processing resource identifier is an integer processing unit number.

1441. The apparatus of claim 1431, wherein the execution-instruction signal includes an address.

1442. The apparatus of claim 1429, wherein an other processing resource may be an delegating processing resource.

1443. The apparatus of claim 1429, wherein a processing resource is an integer processing unit.

1444. The apparatus of claim 1429, wherein a processing resource is a mathematical processing unit.

1445. The apparatus of claim 1429, wherein a processing resource is a memory management unit.

1446. The apparatus of claim 1429, wherein a processing resource is a vector processing unit.

1447. The apparatus of claim 1429, wherein a processing resource is a digital signal processing unit.

1448. The apparatus of claim 1429, wherein a processing resource is a graphics processing unit.

1449. The apparatus of claim 1429, wherein a processing resource is an input/output controller processing unit.

1450. The apparatus of claim 1429, wherein a processing resource is an execution-instruction processing cache.

1451. The apparatus of claim 1429, wherein delegation occurs through an execution-instruction signal router.

1452. The apparatus of claim 1451, wherein the execution-instruction signal router is a cross-point switch.

1453. The apparatus of claim 1429, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

1454. The apparatus of claim 1429, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

1455. The apparatus of claim 1429, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

1456. The apparatus of claim 1429, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

1457. The apparatus of claim 1453, wherein a sleeping processing resource may be woken by a response to a request.

1458. The apparatus of claim 1457, wherein the response is directed to a specific processing resource.

1459. The apparatus of claim 1457, wherein the response indicates a once locked resource is unlocked.

1460. The apparatus of claim 1429, wherein processing resources are communicatively disposed on a same die.

1461. The apparatus of claim 1460, wherein an execution-instruction signal router is on the same die with processing resources.

1462. The apparatus of claim 1429, wherein the determination is made by the instruction processing cache's logic facilities.

1463. The apparatus of claim 1429, wherein the requesting processing resource is unlocked when an unlock instruction is received that specifies the same target memory address.

1464. The apparatus of claim 1429, wherein the target memory address is locked for read and write operations.

1465. The apparatus of claim 1429, wherein the target memory address is locked for read only operations and the request specifies a read and write operation lock.

1466. An apparatus of sharing memory, comprising:

a memory, the memory for storing instructions;

a processing resource that may issue a plurality of processing instructions stored in the memory, wherein the instructions issue signals to:

obtain a response to an execution-instruction from a delegate processing resource, wherein the response includes a target address at a processing resource;

determine if a target address is locked based on a lock variable at the target memory address;

determine value types by using a hash function;

update value types in a primary cache memory of a processing resource to a secondary cache memory for each value type, if each value type in a primary cache memory of a processing resource has not been updated to a secondary cache memory;

1467. The apparatus of claim 1466, wherein the system is completed within a single processing cycle.

1468. The apparatus of claim 1466, wherein the response is an execution-instruction signal for execution by a type of processing resource.

1469. The apparatus of claim 1468, further comprising,

route the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

1470. The apparatus of claim 1468, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

1471. The apparatus of claim 1468, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

1472. The apparatus of claim 1468, wherein the execution-instruction signal includes an operation-code.

1473. The apparatus of claim 1472, wherein the operation-code indicates a type of resource on which to execute.

1474. The apparatus of claim 1468, wherein the execution-instruction signal includes values from a status register.

1475. The apparatus of claim 1468, wherein the execution-instruction signal includes values from a priority bit status register.

1476. The apparatus of claim 1468, wherein the execution-instruction signal includes a processing resource identifier.

1477. The apparatus of claim 1476, wherein the processing resource identifier is an integer processing unit number.

1478. The apparatus of claim 1468, wherein the execution-instruction signal includes an address.

1479. The apparatus of claim 1466, wherein an other processing resource may be an delegating processing resource.

1480. The apparatus of claim 1466, wherein a processing resource is an integer processing unit.

1481. The apparatus of claim 1466, wherein a processing resource is a mathematical processing unit.

1482. The apparatus of claim 1466, wherein a processing resource is a memory management unit.

1483. The apparatus of claim 1466, wherein a processing resource is a vector processing unit.

1484. The apparatus of claim 1466, wherein a processing resource is a digital signal processing unit.

1485. The apparatus of claim 1466, wherein a processing resource is a graphics processing unit.

1486. The apparatus of claim 1466, wherein a processing resource is an input/output controller processing unit.

1487. The apparatus of claim 1466, wherein a processing resource is an execution-instruction processing cache.

1488. The apparatus of claim 1466, wherein delegation occurs through an execution-instruction signal router.

1489. The apparatus of claim 1488, wherein the execution-instruction signal router is a cross-point switch.

1490. The apparatus of claim 1466, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

1491. The apparatus of claim 1466, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

1492. The apparatus of claim 1466, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

1493. The apparatus of claim 1466, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

1494. The apparatus of claim 1490, wherein a sleeping processing resource may be woken by a response to a request.

1495. The apparatus of claim 1494, wherein the response is directed to a specific processing resource.

1496. The apparatus of claim 1494, wherein the response indicates a once locked resource is unlocked.

1497. The apparatus of claim 1466, wherein processing resources are communicatively disposed on a same die.

1498. The apparatus of claim 1497, wherein an execution-instruction signal router is on the same die with processing resources.

1499. The apparatus of claim 1466, wherein the response includes a target processing resource identifier.

1500. The apparatus of claim 1466, wherein the determination is made by a processing resource.

1501. The apparatus of claim 1466, wherein the value type is a global value.

1502. The apparatus of claim 1466, wherein the secondary cache memory is a Level 2 cache memory.

1503. The apparatus of claim 1466, wherein a primary cache memory is a Level 1 cache memory.

1504. The apparatus of claim 1466, further comprising,
marking a cache line of the updated value type as free.

1505. The apparatus of claim 1466, further comprising,
update lock variables in secondary cache memory for each updated value type.

1506. The apparatus of claim 1505, wherein the lock variable updating is performed by the secondary cache memory, which is a processing cache memory.

1507. An apparatus of sharing memory, comprising:

a memory, the memory for storing instructions;

a processing resource that may issue a plurality of processing instructions stored in the memory, wherein the instructions issue signals to:

obtain a storage request including an event variable address and execution-instruction for a processing resource to sleep;

determine if an event variable is set;

provide a reply to a processing resource bus with the event variable address and event value, which will wake an appropriate processing resource from sleep, if the event value is set not to sleep;

set a sleep until event value for the event variable, if the event value is set to sleep.

1508. The apparatus of claim 1507, wherein the system is completed within a single processing cycle.

1509. The apparatus of claim 1507, wherein the storage request is an execution-instruction signal for execution by a type of processing resource.

1510. The apparatus of claim 1509, further comprising,

route the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

1511. The apparatus of claim 1509, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

1512. The apparatus of claim 1509, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

1513. The apparatus of claim 1509, wherein the execution-instruction signal includes an operation-code.

1514. The apparatus of claim 1513, wherein the operation-code indicates a type of resource on which to execute.

1515. The apparatus of claim 1509, wherein the execution-instruction signal includes values from a status register.

1516. The apparatus of claim 1509, wherein the execution-instruction signal includes values from a priority bit status register.

1517. The apparatus of claim 1509, wherein the execution-instruction signal includes a processing resource identifier.

1518. The apparatus of claim 1517, wherein the processing resource identifier is an integer processing unit number.

1519. The apparatus of claim 1509, wherein the execution-instruction signal includes an address.

1520. The apparatus of claim 1507, wherein an other processing resource may be an delegating processing resource.

1521. The apparatus of claim 1507, wherein a processing resource is an integer processing unit.

1522. The apparatus of claim 1507, wherein a processing resource is a mathematical processing unit.

1523. The apparatus of claim 1507, wherein a processing resource is a memory management unit.

1524. The apparatus of claim 1507, wherein a processing resource is a vector processing unit.

1525. The apparatus of claim 1507, wherein a processing resource is a digital signal processing unit.

1526. The apparatus of claim 1507, wherein a processing resource is a graphics processing unit.

1527. The apparatus of claim 1507, wherein a processing resource is an input/output controller processing unit.

1528. The apparatus of claim 1507, wherein a processing resource is an execution-instruction processing cache.

1529. The apparatus of claim 1507, wherein delegation occurs through an execution-instruction signal router.

1530. The apparatus of claim 1529, wherein the execution-instruction signal router is a cross-point switch.

1531. The apparatus of claim 1507, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

1532. The apparatus of claim 1507, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

1533. The apparatus of claim 1507, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

1534. The apparatus of claim 1507, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

1535. The apparatus of claim 1531, wherein a sleeping processing resource may be woken by a response to a request.

1536. The apparatus of claim 1535, wherein the response is directed to a specific processing resource.

1537. The apparatus of claim 1535, wherein the response indicates a once locked resource is unlocked.

1538. The apparatus of claim 1507, wherein processing resources are communicatively disposed on a same die.

1539. The apparatus of claim 1538, wherein an execution-instruction signal router is on the same die with processing resources.

1540. The apparatus of claim 1507, wherein the determination is made at a processing cache.

1541. The apparatus of claim 1507, wherein the event variable is set at the processing cache.

1542. The apparatus of claim 1507, wherein the event variable is set in a register.

1543. The apparatus of claim 1507, wherein the event variable is set in a target address in a processing cache.

1544. An apparatus of sharing memory, comprising:

a memory, the memory for storing instructions;

a processing resource that may issue a plurality of processing instructions stored in the memory, wherein the instructions issue signals to:

obtain a storage request including an event address at a processing cache;

combine a parameter with an event variable through in a processing cache;

provide a reply to a processing resource bus from the combination with the event address, which will wake an appropriate processing resource from sleep that is waiting on an event, if the event value is set not to sleep.

1545. The apparatus of claim 1544, wherein the system is completed within a single processing cycle.

1546. The apparatus of claim 1544, wherein the storage request is an execution-instruction signal for execution by a type of processing resource.

1547. The apparatus of claim 1546, further comprising,

route the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

1548. The apparatus of claim 1546, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

1549. The apparatus of claim 1546, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

1550. The apparatus of claim 1546, wherein the execution-instruction signal includes an operation-code.

1551. The apparatus of claim 1550, wherein the operation-code indicates a type of resource on which to execute.

1552. The apparatus of claim 1546, wherein the execution-instruction signal includes values from a status register.

1553. The apparatus of claim 1546, wherein the execution-instruction signal includes values from a priority bit status register.

1554. The apparatus of claim 1546, wherein the execution-instruction signal includes a processing resource identifier.

1555. The apparatus of claim 1554, wherein the processing resource identifier is an integer processing unit number.

1556. The apparatus of claim 1546, wherein the execution-instruction signal includes an address.

1557. The apparatus of claim 1544, wherein an other processing resource may be an delegating processing resource.

1558. The apparatus of claim 1544, wherein a processing resource is an integer processing unit.

1559. The apparatus of claim 1544, wherein a processing resource is a mathematical processing unit.

1560. The apparatus of claim 1544, wherein a processing resource is a memory management unit.

1561. The apparatus of claim 1544, wherein a processing resource is a vector processing unit.

1562. The apparatus of claim 1544, wherein a processing resource is a digital signal processing unit.

1563. The apparatus of claim 1544, wherein a processing resource is a graphics processing unit.

1564. The apparatus of claim 1544, wherein a processing resource is an input/output controller processing unit.

1565. The apparatus of claim 1544, wherein a processing resource is an execution-instruction processing cache.

1566. The apparatus of claim 1544, wherein delegation occurs through an execution-instruction signal router.

1567. The apparatus of claim 1566, wherein the execution-instruction signal router is a cross-point switch.

1568. The apparatus of claim 1544, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

1569. The apparatus of claim 1544, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

1570. The apparatus of claim 1544, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

1571. The apparatus of claim 1544, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

1572. The apparatus of claim 1568, wherein a sleeping processing resource may be woken by a response to a request.

1573. The apparatus of claim 1572, wherein the response is directed to a specific processing resource.

1574. The apparatus of claim 1572, wherein the response indicates a once locked resource is unlocked.

1575. The apparatus of claim 1544, wherein processing resources are communicatively disposed on a same die.

1576. The apparatus of claim 1575, wherein an execution-instruction signal router is on the same die with processing resources.

1577. The apparatus of claim 1544, wherein the parameter is a bit-mask.

1578. The apparatus of claim 1544, wherein the combination is a logical-OR.

1579. An apparatus of sharing memory, comprising:

a memory, the memory for storing instructions;

a processing resource that may issue a plurality of processing instructions stored in the memory, wherein the instructions issue signals to:

obtain a storage request including a semaphore address at a processing cache;

set a semaphore variable in a processing cache at the semaphore address;

provide an operating system trap-call to wait on a processing queue, if the storage request includes a wait-on-semaphore instruction;

provide an operating system trap-call to signal on a processing queue, if the storage request includes a signal-on-semaphore instruction;

1580. The apparatus of claim 1579, wherein the system is completed within a single processing cycle.

1581. The apparatus of claim 1579, wherein the storage request is an execution-instruction signal for execution by a type of processing resource.

1582. The apparatus of claim 1581, further comprising,

route the execution-instruction signal to an other processing resource, if an operation-code within the execution-instruction signal is for an other processing resource.

1583. The apparatus of claim 1581, wherein the execution-instruction signal is obtained at an execution-instruction signal router.

1584. The apparatus of claim 1581, wherein the execution-instruction signal is provided by a delegating processing resource via instruction-instruction signal router.

1585. The apparatus of claim 1581, wherein the execution-instruction signal includes an operation-code.

1586. The apparatus of claim 1585, wherein the operation-code indicates a type of resource on which to execute.

1587. The apparatus of claim 1581, wherein the execution-instruction signal includes values from a status register.

1588. The apparatus of claim 1581, wherein the execution-instruction signal includes values from a priority bit status register.

1589. The apparatus of claim 1581, wherein the execution-instruction signal includes a processing resource identifier.

1590. The apparatus of claim 1589, wherein the processing resource identifier is an integer processing unit number.

1591. The apparatus of claim 1581, wherein the execution-instruction signal includes an address.

1592. The apparatus of claim 1579, wherein an other processing resource may be an delegating processing resource.

1593. The apparatus of claim 1579, wherein a processing resource is an integer processing unit.

1594. The apparatus of claim 1579, wherein a processing resource is a mathematical processing unit.

1595. The apparatus of claim 1579, wherein a processing resource is a memory management unit.

1596. The apparatus of claim 1579, wherein a processing resource is a vector processing unit.

1597. The apparatus of claim 1579, wherein a processing resource is a digital signal processing unit.

1598. The apparatus of claim 1579, wherein a processing resource is a graphics processing unit.

1599. The apparatus of claim 1579, wherein a processing resource is an input/output controller processing unit.

1600. The apparatus of claim 1579, wherein a processing resource is an execution-instruction processing cache.

1601. The apparatus of claim 1579, wherein delegation occurs through an execution-instruction signal router.

1602. The apparatus of claim 1601, wherein the execution-instruction signal router is a cross-point switch.

1603. The apparatus of claim 1579, wherein a processing resource may sleep while an other processing resource executes delegated execution-instructions.

1604. The apparatus of claim 1579, wherein the execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

1605. The apparatus of claim 1579, wherein the execution-instruction signal from processing resources themselves shuts off processing resources while idling.

1606. The apparatus of claim 1579, wherein the execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

1607. The apparatus of claim 1603, wherein a sleeping processing resource may be woken by a response to a request.

1608. The apparatus of claim 1607, wherein the response is directed to a specific processing resource.

1609. The apparatus of claim 1607, wherein the response indicates a once locked resource is unlocked.

1610. The apparatus of claim 1579, wherein processing resources are communicatively disposed on a same die.

1611. The apparatus of claim 1610, wherein an execution-instruction signal router is on the same die with processing resources.

1612. The apparatus of claim 1579, wherein the trap-call causes rescheduling so that other processes can run on a particular processing resource.

1613. The apparatus of claim 1579, wherein other processes include threads.

1614. The apparatus of claim 1579, wherein other processes include threads.